

Heiko Koch

Konturverfolgung mit Industrierobotern

Heiko Koch

Konturverfolgung mit Industrierobotern

Fusion von Bildverarbeitung, Kraft- und
Beschleunigungssensorik



TECHNISCHE UNIVERSITÄT
CHEMNITZ

Universitätsverlag Chemnitz
2013

Impressum

Bibliografische Information der Deutschen Nationalbibliothek

Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Angaben sind im Internet über <http://dnb.d-nb.de> abrufbar.

Technische Universität Chemnitz/Universitätsbibliothek

Universitätsverlag Chemnitz

09107 Chemnitz

<http://www.bibliothek.tu-chemnitz.de/UniVerlag/>

Herstellung und Auslieferung

Verlagshaus Monsenstein und Vannerdat OHG

Am Hawerkamp 31

48155 Münster

<http://www.mv-verlag.de>

ISBN 978-3-941003-85-9

<http://nbn-resolving.de/urn:nbn:de:bsz:ch1-qucosa-113208>

Konturverfolgung mit Industrierobotern

Fusion von Bildverarbeitung, Kraft- und Beschleunigungssensorik

von der Fakultät für Elektrotechnik und Informationstechnik
der Technischen Universität Chemnitz

genehmigte
Dissertation
zur Erlangung des akademischen Grades
Doktor-Ingenieur
Dr.-Ing.

vorgelegt
von

Dipl.-Ing. (FH) Heiko Koch
geboren am 9. Mai 1982 in Hannover, Deutschland

eingereicht am 7. Februar 2013

Gutachter: Prof. Dr.-Ing. Jozef Suchý
Prof. Dr.-Ing. Wolfgang Weber

Tag der Verleihung: 12. April 2013

Technische Universität Chemnitz
Fakultät für Elektrotechnik und Informationstechnik

Bibliographische Beschreibung:

Diese Dissertation stellt die sensorgeführte Regelung von Industrierobotern zur Konturverfolgung vor. Sensorinformationen aus Kraft-Momenten Sensoren, Kameras und Beschleunigungssensoren werden zusammen mit den Daten eines positionsgeregelten Industrieroboters vereint. Die Bewegung zur Verfolgung von sichtbaren Konturen mit einem robotergeführten Werkzeug wird unter Einhaltung von gewünschten Kontaktkräften vorgegeben (z.B. zum Auftrag von Klebmasse oder beim Nähen von Verbundwerkstoffen). Bei gleichzeitiger Kraftregelung und bildgestützter Regelung entstehen Verkopplungen, welche in dieser Arbeit untersucht werden, um die Regelung auf Basis unterschiedlicher Sensorsysteme unabhängig voneinander durchzuführen (Entkopplung). Darüber hinaus werden Fortschritte bei der Positionsmessung des robotergeführten Werkzeuges gezeigt, indem Beschleunigungssensoren am Werkzeug des Roboters eingesetzt werden.

Schlagwörter

- Industrieroboter
- Konturverfolgung
- Entkopplung
- Laser-Triangulations-Sensor
- Kraft-Momenten-Sensor
- Beschleunigungsmessung
- Sensorfusion
- Beobachter
- Identifikation

Vorwort

Die vorliegende Arbeit entstand während meiner Tätigkeit im Zentrum für Robotik der Hochschule Darmstadt.

Zunächst möchte ich Herrn Prof. Dr.-Ing. Jozef Suchý von der TU Chemnitz sehr herzlich dafür danken, dass er mir die Möglichkeit gegeben hat diese Arbeit unter seiner Betreuung zu erstellen. Er hat durch die Einschätzung der Inhalte und durch seine große Unterstützung sehr zum Gelingen dieser Arbeit beigetragen.

Mein weiterer Dank gilt Herrn Prof. Dr.-Ing. Wolfgang Weber von der Hochschule Darmstadt für die freundliche Übernahme des Korreferats. Ihm danke ich für die kritische Auseinandersetzung mit der Arbeit und den daraus resultierenden Ideen und Anregungen.

Frau Prof. Dr.-Ing. Alexandra Weigl-Seitz von der Hochschule Darmstadt danke ich für die Unterstützung und die vielen zielführenden Diskussionen. Sie stand mir bei vielen kritischen Punkten und Entscheidungen bei, und war immer ein wertvoller Ansprechpartner und Befürworter meiner Arbeit.

Bei Herrn Prof. Dr.-Ing. Karl Kleinmann von der Hochschule Darmstadt möchte ich mich für das große Interesse, die Begeisterung und die Motivation bedanken, welche einen wertvollen Ansporn geleistet haben.

Meinem Kollegen Alexander König danke ich herzlich für anregende Diskussionen und die Hilfe bei Messungen und Aufbauten im Labor.

Mein Dank gilt auch der Hochschule Darmstadt und dem Forschungsprojekt „Hybrino“ des Bundesministeriums für Bildung und Forschung, deren finanzielle Unterstützungen die Durchführung meiner Arbeit ermöglichten.

Ich danke meiner Familie und Freunden für die Aufmunterung und den Rückhalt. Ganz besonders möchte ich meiner Ehefrau Dr. Lisa Koch für das Korrekturlesen danken, und dafür dass sie mich fortwährend ermutigt hat und immer an mich glaubt. Ich danke meinem Sohn Moritz dafür, dass er mich zwei Tage vor der Verteidigung mit seiner Geburt motiviert hat.

Inhaltsverzeichnis

| | |
|--|-----------|
| Abkürzungen und Symbole | 6 |
| Kurzfassung | 8 |
| Abstract | 9 |
| 1 Einleitung | 11 |
| 1.1 Motivation | 11 |
| 1.2 Inhalt und Aufbau der Arbeit | 13 |
| 1.3 Stand der Technik | 16 |
| 1.3.1 Visual Servoing | 16 |
| 1.3.2 Kraftregelung | 25 |
| 1.3.3 Parallele Kraft- und Bildregelung | 30 |
| 1.3.4 Konturverfolgung | 33 |
| 1.4 Systemaufbau | 36 |
| 2 Ermittlung der kartesischen Roboterposition | 43 |
| 2.1 Zusammenfassung | 52 |
| 3 Bildgestützte Konturverfolgung | 55 |
| 3.1 Bildverarbeitung und Messwertspeicherung | 57 |
| 3.1.1 3D-Messprinzip | 58 |
| 3.1.2 Messwertaufzeichnung bei veränderlicher Vorlauflänge | 61 |
| 3.1.3 Geometrische Sortierung der Messwerte | 63 |
| 3.2 Algorithmus zur prädiktiven Konturverfolgung | 67 |
| 3.2.1 Berechnung rotatorischer Stellwerte im Vorlauf . . . | 68 |
| 3.2.2 Prädiktion zur Selektion des Stellwertes | 72 |
| 3.2.3 Berechnung translatorischer Stellwerte | 75 |
| 3.2.4 Filterung des Merkmalspfades | 77 |
| 3.2.5 Automatische Anpassung der Verfahrensgeschwindigkeit | 83 |
| 3.2.6 Erkennung von Ecken | 85 |
| 3.3 Zusammenfassung | 90 |

| | | |
|----------|---|------------|
| 4 | Kraftregelung | 91 |
| 4.1 | Positionsbasierte Kraftregelung | 93 |
| 4.2 | Ermittlung der Kontaktkraft | 94 |
| 4.2.1 | Gravitationskräfte und -momente | 95 |
| 4.2.2 | Dynamische Kräfte und Momente | 96 |
| 4.3 | Beschleunigungsmessung | 97 |
| 4.4 | Kompensation von Trägheitskräften | 101 |
| 4.4.1 | Berechnung der Trägheitskraft aus Roboterdaten . . | 102 |
| 4.4.2 | Berechnung der Trägheitskraft aus Beschleunigungs- sensoren | 104 |
| 4.5 | Zusammenfassung | 106 |
| 5 | Konturverfolgung durch Fusion von Kraft- und Bildsenso- rik | 107 |
| 5.1 | Einfluss der Objektverformung auf die Konturverfolgung . . | 109 |
| 5.2 | Kompensation der Objektverformung | 113 |
| 5.2.1 | Statische Kompensation der Kontaktkraft | 113 |
| 5.2.2 | Filterung und Zeitverzögerung zur Kompensation . . | 119 |
| 5.2.3 | Dynamische Kompensation der Kontaktkraft | 121 |
| 5.2.4 | Identifikation der Umgebungsparameter | 123 |
| 5.3 | Gleichzeitige Kraft- und Bildregelung | 126 |
| 5.3.1 | Experiment 1: Kraftvorsteuerung aus Bildinforma- tionen | 127 |
| 5.3.2 | Experiment 2: Bildgestützte Regelung und Kraftre- gelung in einer Achsrichtung | 132 |
| 5.4 | Zusammenfassung | 134 |
| 6 | Zusammenfassung und Ausblick | 137 |
| A | Dreidimensionale lineare Regression | 141 |
| B | Quadratische Regression | 144 |
| C | Identifikation | 145 |
| D | Bilineare Transformation | 148 |
| | Literaturverzeichnis | 153 |

Abkürzungen und Symbole

| | |
|------------------------------|---|
| 3D, 2D | Dreidimensional, zweidimensional |
| CAD | Computer-Aided Design |
| CCD | Charge-Coupled Device |
| DOF | Degree Of Freedom |
| EOL | Endpoint Open-Loop |
| ECL | Endpoint Closed-Loop |
| IBVS | Image-Based Visual Servoing |
| KMS | Kraft-Momenten-Sensor |
| PBVS | Position-Based Visual Servoing |
| RBO | Rigid-Body Observer |
| RSI | Robot Sensor Interface |
| TCP | Tool Center Point |
| $K_{[TCP]}$ | Werkzeugkoordinatensystem des Roboters |
| $K_{[base]}$ | Basiskoordinatensystem des Roboters |
| \mathbf{T}_a^b | Homogene Transformation vom Koordinatensystem a ins Koordinatensystem b |
| \mathbf{p}^a | Beschreibung des punktes \mathbf{p} im Koordinatensystem a |
| γ, β, α | Rotation um die x -, y - bzw. z -Achse |
| T_{IPO} | Interpolationstakt des Roboters |
| \mathbf{r}, \mathbf{R} | Roboterpose, Roboterpfad |
| \mathbf{a} | Beschleunigung des TCP |
| \mathbf{p}, \mathbf{P} | 3D-Bildmerkmal, Merkmalspfad |
| \mathbf{s}, \mathbf{S}^* | Stellwert für den Roboter, Stellwertpfad |
| \mathbf{F}, \mathbf{M} | Kontaktkraft, Kontaktmoment |
| $\mathbf{F}_d, \mathbf{e}_F$ | Sollwert der Kraft, Regelfehler der Kraft |
| v_x | Bewegungsgeschwindigkeit bei Konturverfolgung |
| d_y, d_z | Sollwert des Abstandes zwischen Werkstück und Kontur |
| $v_{ff,z}$ | Vorsteuergeschwindigkeit in z -Richtung |
| τ_{IP} | Zeitverzögerung durch Bildverarbeitung |
| τ_{fil} | Zeitverzögerung durch Filterung |

| | |
|--------------------|---|
| K, M, D | Federsteifigkeit, Masse, Dämpfung des Werkstückes |
| l_{cam} | Vorlauflänge der Kamera |
| \vec{t}_{adv} | Tangente im Vorlauf |
| $\hat{\mathbf{p}}$ | Prädizierte zukünftige Position des Roboters |

Kurzfassung

Diese Dissertation befasst sich mit der sensorgeführten Regelung von Industrierobotern zur Konturverfolgung. Beispiele dafür sind das robotergetstützte Nähen, Entgraten oder das Auftragen von Dichtmasse entlang von Schweißnähten. Beim Nähen und Entgraten müssen während der Verfolgung der Kontur bestimmte Kontaktkräfte an möglicherweise nachgiebigen Werkstücken eingehalten werden. Dabei ist es in modernen Fertigungsprozessen wichtig, die Bewegung des Roboters mit wenig Einrichtaufwand vorzugeben. Dazu werden Sensorsysteme eingesetzt, die Bildinformationen und Kraftmessungen verarbeiten, um den Roboter mit gewünschter Kontaktkraft entlang sichtbarer Konturen eines Werkstückes zu führen. Der Fokus dieser Arbeit ist die Fusion der Sensordaten, um die Vorteile der einzelnen Sensoren in einer Aufgabe zu vereinen. Es werden Messwerte eines Kraft-Momenten Sensors, einer Kamera, eines Beschleunigungssensors und der kartesischen Position und Orientierung des Roboters verwendet.

Zuerst wird die Berechnung der kartesischen Roboterposition untersucht. Es wird ein Beobachter vorgestellt, um unter Verwendung eines Beschleunigungssensors die Präzision des Positionssignales zu erhöhen. Anschließend wird das Kamerasystem untersucht und ein Verfahren vorgestellt, um Geschwindigkeit, Position und Orientierung des robotergeführten Werkzeuges entlang der Kontur vorzugeben. Danach wird auf die Ermittlung von Kontaktkräften eingegangen, wobei die Kompensation von Trägheitskräften mittels Beschleunigungssensoren untersucht wird. Der letzte Abschnitt befasst sich mit der Verbindung von bildgestützter Konturverfolgung und Kraftregelung an nachgiebigen Werkstücken. Durch die Nachgiebigkeit des Werkstückmaterials verformt sich die Kontur bei Kontakt. Durch bildgestützte Konturverfolgung wird eine Anpassung an diese Verformung vorgenommen – somit besteht über die Verformbarkeit des Werkstückes eine Kopplung zwischen den beiden Regelkreisen. Diese Kopplung wird gelöst, indem auf Basis eines dynamischen Modells der Umgebung eine Kompensation der Werkstückverformung berechnet wird. Die Modellparameter zur Kompensation werden durch online Identifikation ermittelt.

Abstract

This thesis focuses on the sensor-guided control of industrial robots for contour-following. Examples include the robot-guided sewing, grinding or the application of sealant along weld seams. Grinding and sewing require a certain contact force while following a contour of a workpiece, whereas the workpieces might be compliant. Modern production processes require a fast and simple way to set up the motion of the robot for the required task. Therefore sensor systems are used, which process visual and tactile information to guide the tool at a desired contact force along visible contours of a workpiece. The focus of this work is the fusion of sensor data, used to benefit from the advantages of each of the individual sensors in one control scheme. I combine the measurements of a force-torque sensor, a camera, an acceleration sensor and of the Cartesian position of the robot.

First, I introduce details on the calculation of the Cartesian robot position. I present an observer-based structure that uses an acceleration sensor to improve the precision of the robot position signal. Then, I analyze the camera system and present a control structure that adapts the position, orientation and velocity of the robot-guided tool along the contour. Thereafter, I show details of force measurement, whereas I compensate for inertial forces using an acceleration sensor. The last chapter addresses the combination of visual contour-following and force control on compliant workpieces. Under a certain contact force, the workpiece deforms due to its compliance. The position and orientation then is adapted to this deformed contour by visual control – hence, there is a coupling between force and visual control. This coupling is solved by compensating for workpiece deformation using a dynamic model of the environment. The environmental parameters for compensation are identified online.

1 Einleitung

Diese Einleitung gibt einen Überblick über die Zielsetzung und Einsatzgebiete von Konturverfolgung in der Robotik. Der erste Abschnitt beschreibt die Motivation dieser Arbeit. Im Anschluss folgen der Inhalt und Aufbau der Arbeit sowie der aktuelle Stand der Technik auf dem Gebiet der untersuchten Teilgebiete. Der Systemaufbau zur experimentellen Validierung der vorgestellten Algorithmen wird im letzten Abschnitt erläutert.

1.1 Motivation

Roboter sind aus modernen Fertigungsanlagen nicht mehr wegzudenken. Sie zeichnen sich gegenüber manuell durchgeführten Prozessen durch sehr hohe Wiederholgenauigkeiten bei pausenlos hohen Geschwindigkeiten aus. Bei manueller Produktion birgt eine extrem monotone Aufgabe die Gefahr unbemerkter Fehler durch Ermüdung des Personals. Diese Ermüdungserscheinungen treten bei einem Robotersystem nicht auf. Zudem werden Personalkosten eingespart. Bei konstanter Verarbeitungsdauer und gleichbleibender Qualität eines Produktionsschrittes sind mit Robotern hohe Arbeitstakte erreichbar. Diese tragen durch die verbesserte Auslastung einer Anlage zur Senkung von Produktionskosten bei.

Häufig werden Roboter jedoch nur als reine Bewegungsapparate mit hoher Wiederholgenauigkeit ohne Erfassung der Umwelt eingesetzt. In diesem Fall muss die Umgebung des Roboters stark strukturiert und bekannt sein, d.h., sowohl die Position als auch die Form eines Werkstückes müssen vor Ausführung einer Aufgabe vollständig bekannt sein und während des Prozesses konstant bleiben. Ein enormer zeitlicher Aufwand und Expertenwissen sind zur Einrichtung des Prozesses notwendig und nur ökonomisch, wenn sehr große Stückzahlen produziert werden. Die heutzutage immer weiter steigenden Fähigkeiten von Sensoren (Kraftsensoren, bildgebende Sensoren, Beschleunigungssensoren, Temperatursensoren, ...) und die stei-

gende Rechenleistung von Computern bei sinkenden Kosten ermöglichen steigende Autonomie von Systemen, sodass der Anspruch an die strukturierte Umgebung sinken kann. Durch den Einsatz von Sensoren kann das Robotersystem die Umgebung wahrnehmen. Durch Bild- und Kraftsensorik soll der Roboter „sehen“ und „fühlen“ können, um z.B. Objekte aus einer unsortierten Kiste zu greifen und beim Zusammenfügen von Objekten nachgiebig zu reagieren.

Ein Beispiel für bildgestützte Verfahren ist die Applikation von Dichtmasse entlang der Schweissnähte einer Karosserie in der Automobilherstellung. Die große Modellvielfalt und Kundenwünsche sorgen für eine Vielfalt an Konturen, entlang welcher der Roboter die Dichtmasse aufzutragen hat. Um die gewünschte Trajektorie¹⁾ des Roboters zu definieren, ist je nach Komplexität ein Zeitaufwand von einigen Stunden bis Tagen notwendig. Der Einsatz von Kameras zur Führung des Roboters kann den Einrichtungsaufwand erheblich reduzieren, da lediglich einige Randbedingungen spezifiziert werden müssen. Die eigentliche Verfolgung einer beliebig geformten Kontur erfolgt daraufhin autonom.

Weitere Anwendungsbeispiele wie z.B. das robotergestützte Nähen von Kunststoffen oder Faserverbundstoffen erfordert neben der Führung des Roboters entlang einer Werkstückkontur die Überwachung bzw. Regelung der Kontaktkraft zwischen Werkstück und Roboter. Was bei manuellen Prozessen nach Gefühl erfolgt, bedeutet für einen Roboter einen erhöhten Einsatz von Sensorik, mit der die Kontaktkraft ermittelt und während des Prozesses eingehalten werden muss. Besonders dann, wenn sich während eines Prozesses mit Umgebungskontakt die Umgebung zusätzlich durch den Kontakt verformt, sind erhöhte Anforderungen an die Interpretation von Sensorsignalen erforderlich, da die Signale aus unterschiedlichen Sensoren für einen Prozess ideal miteinander verknüpft werden müssen (Sensorfusion). Bei der Sensorfusion sollen sowohl entstandene Kopplungen zwischen komplementären Sensorsystemen erkannt und berücksichtigt werden, als auch unterschiedliche Messmethoden miteinander verknüpft werden, um deren Vorteile in einem Gesamtsystem zu vereinen.

Diese Arbeit soll einen Beitrag dazu leisten, die Autonomie von Robotersystemen zu erhöhen, um Konturen mit und ohne Kontakt an nachgiebigen Ojekten zu verfolgen. Die bildgestützte Führung mittels Laser-

¹⁾Eine Trajektorie bezeichnet die Abfolge von Posen entlang eines gewünschten Pfades im Raum mit zeitlicher Zuordnung. D.h., es werden neben dem geometrischen Pfad die Geschwindigkeit (und Beschleunigung) mit berücksichtigt [SSVO09].

Triangulationssensor wird gezeigt, und es wird auf die Fusion von Kraft-, Bild- und Beschleunigungssensorik zur Führung eines Roboters eingegangen. Der Fokus der Sensorfusion liegt neben der Kompensation von Trägheitskräften in der Kraftmessung auf der Entkopplung der bildgestützten Regelung von der Kraftregelung an nachgiebigen Objekten. Es werden Algorithmen und Experimente gezeigt, welche sich auf allgemeine Robotersysteme anwenden lassen.

1.2 Inhalt und Aufbau der Arbeit

In dieser Arbeit wird ein Gesamtkonzept zur Konturverfolgung mit Industrierobotern an Konturen mit unbekanntem Verlauf vorgestellt. Ziel ist es, einer visuell erfassbaren Kontur auf einem beliebig geformten Werkstück zu folgen, wobei neben der Position entlang der Kontur auch die Orientierung und die Geschwindigkeit berücksichtigt werden sollen. Zusätzlich wird Objektkontakt berücksichtigt, wobei der Fokus auf nachgiebige Werkstücke gelegt wird. Kopplungen zwischen bildbasierter Regelung und Kraftregelung werden untersucht. Abbildung 1.1 gibt eine Übersicht über die Struktur der Arbeit mit Kapitelangaben. Die Arbeit ist in vier Schwerpunkte unterteilt:

1. Ermittlung der Roboterposition, Kapitel 2.

Industrieroboter stellen über Sensorschnittstellen die aktuelle kartesische Position des Tool-Center-Points (TCP) zur Verfügung. Abhängig vom Gelenkwinkel der sechs Armeile wird mit der Vorwärtstransformation des Roboters die aktuelle Position berechnet. Da die Gelenkwinkel nicht direkt im Gelenk, sondern über Getriebefaktoren auf der Motorwelle gemessen werden, ergeben sich, hauptsächlich bei Beschleunigungen, Fehler. Um die Genauigkeit der Positionsmessung zu erhöhen, wird ein Beobachter vorgestellt, welcher neben dem herkömmlichen Positionssignal einen am Roboter-Handflansch montierten 3-Achs Beschleunigungssensor berücksichtigt.

2. Prädiktive²⁾ bildbasierte Regelung zur Konturverfolgung, Kapitel 3 [KKK⁺11c, KKK⁺12].

Mittels eines Laser-Triangulationssensors werden Merkmale entlang der zu verfolgenden Kontur gemessen. Diese Merkmale werden nicht

²⁾Die Prädiktion wird nicht im Sinne der klassischen Regelungstechnik betrachtet (z.B. [Smi59, CMT87]), sondern als Bahnprädiktion durch vorausschauende Sensorik.

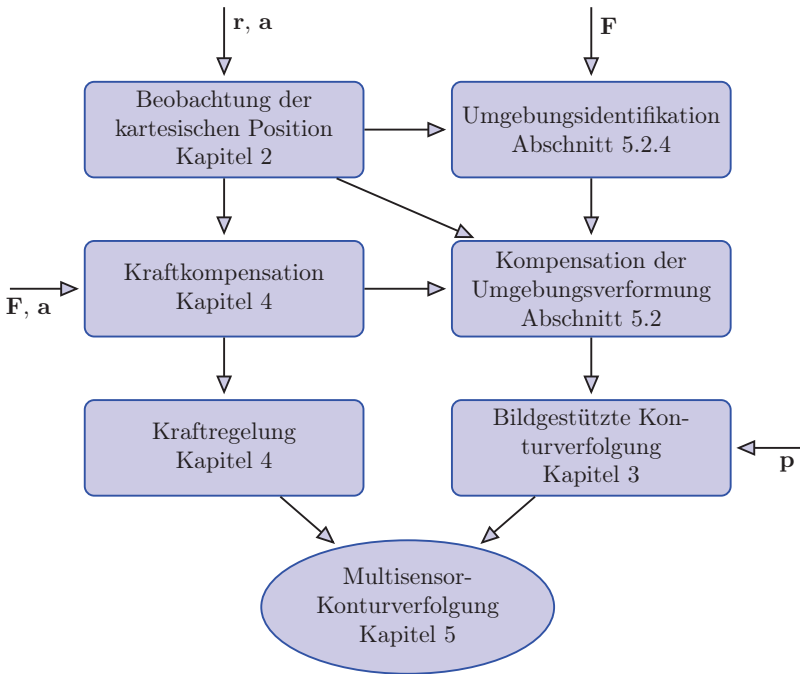


Abbildung 1.1: Übersicht über die Arbeit mit Kapitelangabe. Die Pfeile beschreiben den Signalfluss zwischen den Komponenten. Die Signale entsprechen den Sensorsignalen aus Kraftmessung \mathbf{F} , Beschleunigungsmessung \mathbf{a} , Kamerainformation \mathbf{p} und Position und Orientierung \mathbf{r} des Roboters.

direkt am TCP gemessen, sondern mit einem Vorlauf vor dem TCP. Die folgenden Eigenschaften werden berücksichtigt:

- Filterung, Sortierung und Vorverarbeitung von 3D-Bilddaten. Flexible Anordnungen und variable Sensorvorlängfe werden durch die Sortierung ermöglicht.
- Verfolgung einer 3D-Kontur mit dem Roboterwerkzeug unter Berücksichtigung der Orientierung.
- Vorausschauende Berechnung kartesischer Geschwindigkeiten mit automatischer Anpassung der Verfahrensgeschwindigkeit an die Krümmung der Kontur im Vorlauf.
- Erkennen und Berücksichtigen von Ecken in der Kontur.

3. **Kraftregelung (Kraftmessung), Kapitel 4** [KKWSK10].

Um eine Kraftregelung zu implementieren, ist die Kenntnis der aktuellen Kontaktkraft zwischen Werkzeug und Werkstück notwendig. Die Messung eines Kraftsensors beinhaltet jedoch zusätzlich Trägheitskräfte, Gravitationskräfte, Zentripetalkräfte und Corioliskräfte, welche aus der Messung kompensiert werden müssen. In diesem Kapitel wird diese Kompensation vorgestellt. Es wird das Signal eines Beschleunigungssensors verwendet, um dynamische Einflüsse der Roboterbewegung zu erfassen. Kommerzielle 12DOF (Degrees of Freedom)³⁾ Sensoren werden mit Low-Cost Beschleunigungssensoren hinsichtlich der Signalqualität verglichen. Die Positionierung der Sensoren auf dem Roboterwerkzeug wird untersucht.

4. **Kombinierte Kraft- und Bildregelung, Kapitel 5** [KKK⁺11a, KKK⁺11b].

Gleichzeitiger Einsatz von bildgestützter Regelung und Kraftregelung, insbesondere an nachgiebigen Materialien, erzeugt aufgrund von Verformung des Werkstückes Kopplungen zwischen der bildgestützten Konturverfolgung und der Kraftregelung. Eine Entkopplung der Regelkreise durch Kompensation der Verformung wird vorgestellt, indem ein Modell der Umgebung erstellt wird, welches die Steifigkeit, Masse und Dämpfung berücksichtigt. Die notwendigen Parameter werden mittels Identifikation ermittelt. Die Verbesserung der gleichzeitigen bildgestützten Regelung mit Kraftregelung durch die Entkopplung wird an Beispielen vorgestellt.

³⁾Die Bezeichnung „12DOF“ bedeutet, dass Kräfte und Momente in allen sechs Raumrichtungen sowie die sechs translatorischen und rotatorischen Beschleunigungen in allen sechs Raumrichtungen erfasst werden.

1.3 Stand der Technik

Das Themengebiet der Konturverfolgung mit Industrierobotern an nachgiebigen Objekten kann grundsätzlich in die drei folgenden Themengebiete unterteilt werden:

- Kraftregelung
- Bildgestützte Konturverfolgung (Visual Servoing)
- Parallele Kraft- und Bildregelung

Entsprechend dieser drei Themengebiete werden in diesem Abschnitt der Stand der Technik erläutert und Klassifizierungen aufgezeigt, um die vorliegende Arbeit in der aktuellen Forschung einzuordnen.

1.3.1 Visual Servoing

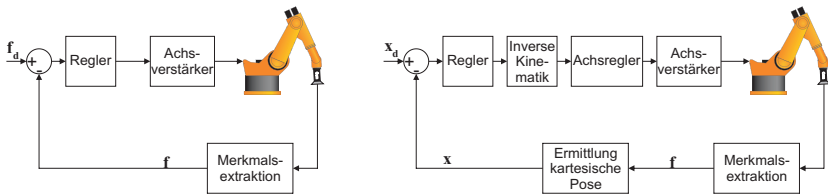
Der Begriff „Visual Servoing“ wurde Mitte der 1980er Jahre maßgeblich von Sanderson und Weiss geprägt [WSN87, WSN85]. Er beschreibt eine Regelungsstruktur, welche einen Roboter und ein Kamerasystem in einem geschlossenen Regelkreis verwendet, um den Roboter zu einem sichtbaren Merkmal zu führen. Heutzutage wird der Begriff weitläufiger für alle Systeme mit bildgestützter Führung verwendet. Besonders seit 1990 stand Visual Servoing im Fokus der Forschung. Detaillierte Übersichten über Verfahren des Visual Servoing wurden z.B. von Corke [Cor94], Hutchinson et al. [HHC96], Hashimoto [Has03] und Chaumette et al. [CH06] veröffentlicht.

Allgemein stellt sich Visual Servoing der Aufgabe, einen Roboter bzw. ein mechanisches System mithilfe bildgestützter Sensoren zu führen. Die Aufgabe wird durch die Minimierung des Fehlers

$$\mathbf{e}(t) = \mathbf{s}(\mathbf{f}(t), \mathbf{a}) - \mathbf{s}^* \quad (1.1)$$

definiert. Dabei ist $\mathbf{f}(t)$ ein Messwert in Bildkoordinaten. \mathbf{a} beinhaltet erweitertes Systemwissen, wie z.B. extrinsische oder intrinsische Kameraparameter⁴⁾, um den aktuellen Messwert \mathbf{s} im gewünschten Koordinatensystem zu beschreiben. \mathbf{s}^* ist der (hier nicht zeitveränderliche) Sollwert der

⁴⁾Kameraparameter werden aus einer Kalibrierung ermittelt. Zu den intrinsischen Parametern zählen die Geometrie und die optischen Eigenschaften von Kamera und Objektiv, zu den extrinsischen Parametern zählen Position und Orientierung der Kamera bezüglich eines bestimmten Weltkoordinatensystems [Tsa87].



(a) **IBVS**: Image-Based Visual Servoing. (b) **PBVS**: Position-Based Visual Servoing.

Abbildung 1.2: Visual Servoing Struktur (direct visual control) nach Weiss [WSN87, WSN85]. Der Regelkreis wird ausschließlich direkt über das Kamerasystem geschlossen.

Bildmerkmale. Die unterschiedlichen Ansätze des Visual Servoing unterscheiden sich hauptsächlich in der Art, wie \mathbf{s} und \mathbf{s}^* beschrieben werden. Dementsprechend werden die Ansätze nachfolgend klassifiziert.

Klassifizierung bildgestützter Regelungsmechanismen

Die Unterscheidung bildgestützter Regelungsmechanismen basiert auf der Klassifizierung von Sanderson und Weiss [WSN87, WSN85] mit den folgenden Definitionen:

- **Visual Servoing** (auch als *direct visual control* bezeichnet): Der gesamte Regelkreis wird, wie in Abbildung 1.2 dargestellt, über die Bildinformation geschlossen, sodass die Gelenkregelung der Robotersteuerung nicht zur Regelung und Stabilisierung der Gelenke verwendet wird. Dabei wird nach bildbasierten und positionsbasierten Verfahren unterschieden:
 - **IBVS** (Image-Based Visual Servoing), Abbildung 1.2(a): Das Bildmerkmal \mathbf{f} ist in Bildkoordinaten direkt aus dem Kamerabild verfügbar. Der Sollwert \mathbf{f}_d wird in Bildkoordinaten definiert. Somit steht auch der Regelfehler $\mathbf{e} = \mathbf{f}_d - \mathbf{f}$ direkt zur Verfügung.
 - **PBVS** (Position-Based Visual Servoing), Abbildung 1.2(b): Aus dem Messwert \mathbf{f} wird das dreidimensionale (kartesische) Merkmal \mathbf{x} berechnet. Eine Kalibrierung der Kamera ist dazu notwendig und beeinflusst die Qualität von \mathbf{x} . Der Sollwert \mathbf{x}_d wird in kartesischen

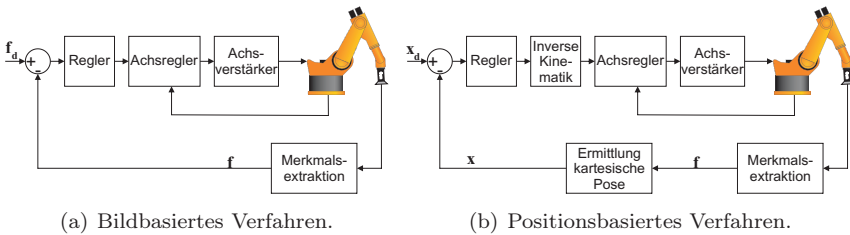


Abbildung 1.3: Dynamic look-and-move Struktur nach Weiss [WSN87, WSN85]. Die Stabilisierung der Gelenke wird durch die Achsregelung der Robotersteuerung übernommen, die Bildverarbeitung generiert Stellwerte für die unterlagerten Achsregler.

Koordinaten vorgegeben, sodass der Regelfehler durch $\mathbf{e} = \mathbf{x}_d - \mathbf{x}$ beschrieben wird.

- **Dynamic Look-and-Move:** Abbildung 1.3 zeigt das Dynamic Look-and-Move Prinzip. Der Regelkreis des Roboters wird nicht direkt über die Bildverarbeitung geschlossen und stabilisiert, sondern über die internen Achsregler der Robotersteuerung. Aus der Bildverarbeitung werden lediglich Stellwerte (z.B. kartesische Sollwerte) für die unterlagerte Achsregelung ermittelt. Auch diese Methode unterteilt sich in positionsbasierte und bildbasierte Verfahren. Dementsprechend wird der Sollwert \mathbf{x}_d bzw. \mathbf{f}_d auf kartesischer Ebene bzw. in Bildkoordinaten vorgegeben. Sehr viele bildgestützte Systeme werden durch die Implementierung von positionsbasiertem Dynamic Look-and-Move realisiert [HHC96, BS04]. Ein Grund dafür ist die häufig niedrige Abtastfrequenz von Bildverarbeitungssystemen. Darüber hinaus verfügen viele Robotersteuerungen ohnehin nur über eine Schnittstelle zur Übermittlung kartesischer Geschwindigkeits- oder Positionsvorgaben. Kinematische Singularitäten werden dadurch in der Robotersteuerung berücksichtigt, sodass der Roboter als idealer kartesischer Bewegungsapparat betrachtet wird. Weiterhin sind Algorithmen durch diese Entkopplung einfacher zwischen unterschiedlichen Systemen portierbar.

Des Weiteren werden Visual Servoing Systeme in Abhängigkeit der Reglerstruktur und Kamerakonfiguration folgendermaßen klassifiziert:

- **Reglerstruktur**

- **EOL** (Endpoint Open-Loop): Nur die Position des Zielobjektes wird zur Generierung des Stellwertes ausgewertet. Eine exakte Kamerakalibrierung ist speziell bei PBVS notwendig, um die Merkmalsposition in kartesischen Koordinaten zu berechnen.
- **ECL** (Endpoint Closed-Loop): Sowohl die Position des Werkzeuges als auch die Position des Zielobjektes werden aus dem Bild extrahiert. Der Regelfehler kann direkt als Abstand zwischen Werkzeug und Zielobjekt aus der Bildaufnahme extrahiert werden, sodass eine ungenaue Kamerakalibrierung keinen Einfluss auf die Genauigkeit des Systems hat.

- **Kamerakonfiguration**

- **Mitgeführte Kamera (eye-in-hand)**: Die Kamera wird am Endeffektor⁵⁾ des Roboters befestigt. Es besteht ein bekannter (meist konstanter) Zusammenhang zwischen der Kamerapose und der Lage des Endeffektors des Roboters.
- **Raumkamera (eye-to-hand)**: Die Kamera wird im Raum montiert. Es besteht ein bekannter (meist konstanter) Zusammenhang zwischen der Kamerapose und der Umgebung bzw. der Roboterbasis.

Chesi [CH02] vergleicht Visual Servoing in Bezug auf die Kamerakonfigurationen eye-in-hand und eye-to-hand. Stabilität wird für beide Konfigurationen sowohl bei IBVS als auch bei PBVS erreicht. Flandin [FCM00] stellt einen Algorithmus vor, welcher die globale Sicht einer feststehenden Raumkamera für translatorische Bewegung mit der lokalen hochauflösenden Sicht einer mitgeführten Kamera für rotatorische Bewegung kombiniert. Somit werden Vorteile der beiden Kamerakonfigurationen in voneinander unabhängigen Richtungen vereint.

Der Vorteil der Regelung in kartesischen Koordinaten (positionsbasierte Verfahren, PBVS) ist, neben der Portierbarkeit von Algorithmen, die Möglichkeit, optimale kartesische Trajektorien im Raum zu generieren. Zur Generierung der Trajektorie werden jedoch Strukturen benötigt, um die Daten in verschiedenen Ebenen zu verarbeiten: Daten aus dem Sensorkoordinatensystem werden in ein Werkstückkoordinatensystem und anschließend auf die Gelenkebene des Roboters transformiert [WSN87]. Die 3D-

⁵⁾Der Endeffektor ist das letzte Element der kinematischen Kette des Roboters, mit welchem die gewünschte Manipulation am Werkstück durchgeführt wird.

Bilddaten müssen mithilfe unterschiedlicher photogrammetrischer Techniken aus der 2D-Projektion der Bildaufnahme gewonnen werden. Somit wird das Visual Servoing System zusätzlich hinsichtlich des verwendeten Kamerasystems klassifiziert:

- **Mono Kamerasystem:** Der einfachste Fall ist die Montage einer einzelnen Kamera mit der optischen Achse senkrecht zur Werkstückebene einer planaren Anordnung. Mithilfe der bekannten Höhe zwischen Kamera und Umgebung wird die dreidimensionale Merkmalsposition ermittelt. Für allgemeinere Anordnungen kann eine bekannte Objektgeometrie dazu verwendet werden, fehlende Tiefeninformationen zu rekonstruieren. Bei Kontakt mit der Umgebung kann bei bekannten Abmessungen des Werkzeuges die fehlende Tiefeninformation im Kontaktpunkt ermittelt werden [BDS01].
- **Stereo Kamerasystem:** Eine Szene wird von zwei Kameras mit bekannter Position aufgezeichnet. In beiden Kamerabildern müssen Merkmale gefunden und einander zugeordnet werden (Korrespondenzproblem) [YC78]. Aus den Merkmalen in beiden Kamerabildern kann die Mehrdeutigkeit der zweidimensionalen Projektion der Szene auf ein Kamerabild gelöst werden, um die Tiefeninformation zu rekonstruieren [PSAR09, KMN⁺93]. Die **Depth from motion** Technik ist einem Stereo Kamerasystem vergleichbar, wobei jedoch die Szene von einer beweglichen Kamera aus unterschiedlichen Blickwinkeln erfasst wird. Das Korrespondenzproblem muss hier ebenso gelöst werden [Nev76].
- **Lasertriangulationsverfahren:** Eine Lichtquelle (z.B. Laser- oder LED-Licht) projiziert über eine Optik einen Lichtstreifen auf das Werkstück. Eine Kamera zeichnet diese Projektion auf. Bei bekannter Relation einer kalibrierten Kamera zum kalibrierten Projektor kann durch das Lichtschnittverfahren (Triangulation) entlang der Projektion im Bild die Höheninformation berechnet werden. Dabei wird entweder eine konstante oder modulierte Lichtquelle verwendet [ZZZJ08, Agi85, MM06]. Kim [KSCK95] beschreibt den Vorteil des engen Frequenzbandes von Laserlicht (z.B. $690nm$ Wellenlänge) beim Einsatz eines optischen Bandpassfilters zur Unterdrückung von Umgebungslicht. Suyi [SLH⁺09] stellt einen Filterungsalgorithmus vor, der Vorwissen, wie z.B. erwartete Helligkeit des Laserstreifens und dessen Dicke, in die Filterung mit einbezieht, um Reflexionen und restliches Umgebungslicht von der gesuchten Laserprojektion zu unterscheiden.

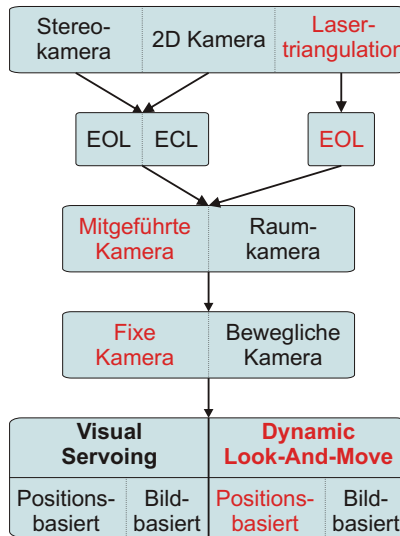


Abbildung 1.4: Klassifizierung bildbasierter Verfahren zur Kommandierung von Robotern (rote Blöcke stehen für die Wahl in dieser Arbeit).

Die Klassifizierung von Systemen zur bildgestützten Führung von Robotern ist in Abbildung 1.4 zusammenfassend dargestellt. Die vorliegende Arbeit wird mit einem **eye-in-hand Lasertriangulationssensor** experimentell validiert. Zwar können Daten nur entlang eines Streifens im Bild ermittelt werden, jedoch ist die Auswertung dieser Bilder sehr schnell und robust gegenüber Störungen (z.B. Lichtschwankung). Ein Stereo Kamerasystem stellt zwar deutlich mehr Informationen zu Verfügung, ist aber auch mit erheblichem Rechenaufwand verbunden. Ebenso wären die Einschränkungen eines Mono Kamerasystems für die Anwendung in dieser Arbeit zu groß. Durch den eingeschränkten Informationsgehalt von Lasertriangulation stellt sich die Reglerstruktur als **EOL** System dar, da die Position des Werkzeuges aus der Bildinformation nicht zu erfassen ist. Aufgrund der eingeschränkten Schnittstelle des verwendeten Roboters wird ein **positionsbasiertes** Verfahren über kartesische Positionsvorgabe eingesetzt. Da das Kamerasystem nicht den Kontaktpunkt beobachtet, sondern entlang der Kontur vorausschauend montiert ist, wird das **Dynamic Look-and-Move Prinzip** zur online Generierung einer Trajektorie für den positionsgeregelten Roboter angewendet. Da der Regelkreis nicht direkt über

das Kamerasystem geschlossen ist, besteht die Möglichkeit, Verzögerungen durch Roboter und Bildverarbeitung adäquat zu berücksichtigen.

Um die Motivation, die zur Auswahl der Architektur in dieser Arbeit geführt hat, zu begründen, werden nachfolgend Eigenschaften und der Stand der Technik zu Methoden des Visual Servoing dargestellt.

Eigenschaften von Visual Servoing

Die Definition des Fehlers $\mathbf{e}(t)$ nach Gleichung 1.1 führt direkt zu dem Entwurf eines Geschwindigkeitsreglers [CH06]. Die Geschwindigkeit \mathbf{v}_c der Kamera (bzw. des Endeffektors) und die Bewegung des Merkmales \mathbf{s} werden in Relation gesetzt (\mathbf{v}_c beinhaltet sowohl rotatorische als auch translatorische Komponenten). Dieses Verhältnis wird durch

$$\dot{\mathbf{s}} = \mathbf{L}_s \mathbf{v}_c \quad (1.2)$$

beschrieben. \mathbf{L}_s ist die Interaktionsmatrix⁶⁾ und enthält unter anderem intrinsische Parameter des Kamerasystems. Sie beschreibt das Verhältnis zwischen Sensor und Umgebung [HHC96, ZZC08]. Aus Gleichung 1.1 und 1.2 folgt bei nicht-zeitveränderlichem Sollwert \mathbf{s}^* direkt der Zusammenhang zwischen Kamerageschwindigkeit und Ableitung des Fehlers

$$\dot{\mathbf{e}} = \mathbf{L}_e \mathbf{v}_c, \text{ mit } \mathbf{L}_e = \mathbf{L}_s \quad (1.3)$$

Das Regelungsziel wird somit durch die Minimierung von \mathbf{e} definiert.

Uncalibrated Visual Servoing

Zur Berechnung der Interaktionsmatrix sind Kameraparameter aus einer Kalibrierung erforderlich. Die Kalibrierung ist jedoch aufwändig und daher mit Kosten verbunden. Ein Forschungsschwerpunkt auf dem Gebiet des Visual Servoing behandelt daher die Thematik, die Interaktionsmatrix online ohne Kalibrierung zu ermitteln (*uncalibrated visual servoing*). Liu et al. [LWWL06, WLZ07, WLZ08] führen eine tiefenunabhängige Interaktionsmatrix ein, um die Regelung in der Bildebene ohne Kenntnis von Kameraparametern durchzuführen. Yoshimi und Allen [YA94] umgehen für eine

⁶⁾Auch die Bezeichnung Bild-Jakobimatrix oder Sensitivity Matrix ist neben der Interaktionsmatrix in der Literatur häufig vertreten [CC00, BS04].

Bolzen-in-Loch Anwendung die notwendige Kalibrierung durch Montage der Kamera parallel zur letzten Achse des Roboters. Dadurch ist bei Rotation der letzten Achse das Ziel im Bild unbewegt, sobald sich der Roboter senkrecht über dem Loch befindet. Espiau [Esp94] untersucht den Einfluss ungenauer Kalibrierung auf das Verhalten von Visual Servoing und zeigt, dass Visual Servoing auch mit ungenauer Kalibrierung Stabilität erhalten kann. Des Weiteren werden adaptive Verfahren vorgeschlagen, um das Verhältnis zwischen Roboterposition (oder Gelenkwinkeln) und Bildbewegung online zu lernen [ZZC08, ATH99].

Positionsbasierte und bildbasierte Verfahren im Vergleich

Bei Verwendung von Kamerabildern sowohl bei IBVS als auch bei PBVS in einem geschlossenen System wird die Bandbreite und somit die Stabilität des Systems beeinträchtigt, da Bildaufnahme und Bildverarbeitung eine deutliche Verzögerung im offenen Regelkreis erzeugen. Zu diesen Verzögerungen gehören die Integrationszeit des Bildsensors, das Auslesen des Sensors und die Datenübertragung sowie die Merkmalsextraktion bei der Bildverarbeitung, welche im System mit berücksichtigt werden müssen [Cor95, CG96]. Diese Problematik kann durch Prädiktion umgangen werden. Sim [SHL02] setzt für eine Objektverfolgungsaufgabe einen Smith-Predictor ein, um die Verzögerung des Bildverarbeitungssystems zu umgehen. Jedoch ist dafür ein möglichst genaues Systemmodell notwendig.

Bildbasiertes Visual Servoing (IBVS) nutzt die Bildinformation (mit Kenntnis der Interaktionsmatrix) als Eingang zur Gelenkebene des Roboters. Der Regelfehler wird in Bildkoordinaten berücksichtigt. Bei ECL wird der relative Abstand (in Pixeln) zwischen dem Endeffektor und der Umgebung geregelt. Bei EOL wird ein bestimmtes „Wunschbild“ vorgegeben (siehe Abbildung 1.5), d.h. bestimmte Merkmale des Kamerabildes müssen sich an einer vorgegebenen Pixelposition befinden⁷⁾ [CH06, ECR92, HKEK91]. Das „Wunschbild“ wird meistens durch Teach-In eingelesen, d.h., es wird ein Bild aufgezeichnet, bei dem der Roboter die korrekte Relation zum Werkstück einnimmt. Bei einer Regelung auf dieses Bildmerkmal wird somit die gewünschte Relation zum Werkstück erreicht.

Im Gegensatz zu IBVS ist die Qualität des Ergebnisses bei PBVS direkt von der Qualität der Kalibrierung abhängig. Das System ist empfindlich

⁷⁾Um 6 Freiheitsgrade zu regeln, sind mindestens 3 Bildmerkmale notwendig [CH06].

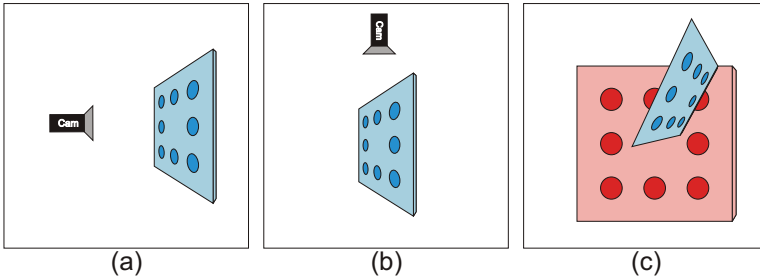


Abbildung 1.5: Beispiel einer bildbasierten Positionieraufgabe: (a) die gewünschte Kamerapose bezüglich des Objektes, (b) die Kamerapose zu Beginn der Aufgabe, (c) das Kamerabild zu Beginn (blau) und nach Abschluss (rot) der Positionierung (nach [CH06]).

gegenüber Kalibrierungsfehlern. Diese Empfindlichkeit kann durch Filterung reduziert werden, z.B. durch Einsatz eines Kalman Filters, welcher modellbasiert die lineare oder zirkuläre Bewegung eines Objektes vorher-sagt [NHNT00].

Deguchi [Deg98] beschreibt die Problematik der meisten klassischen Visual Servoing Ansätze damit, dass Rotation und Translation zwei deutlich unterschiedliche Einflüsse auf die Position von Punkten im Bild haben:

- Eine **Translation** der Kamera verschiebt einen Punkt im Bild auf speichenförmigen Linien. Weiter entfernte Punkte bewegen sich bei Translation deutlich stärker im Bild als nähere Punkte.
- Durch eine **Rotation** der Kamera bewegt sich ein Bildpunkt auf einer Ellipse. Die Ermittlung der Objektorientierung im Bild ist nicht vom Abstand zur Kamera abhängig.

Die übliche gleichzeitige Minimierung dieser beiden Fehler (Rotation und Translation) führt zu ungewollten Bewegungen im kartesischen Raum. Durch Epipolargeometrie zwischen dem Zielbild und dem aktuellen Bild entkoppelt Deguchi Translationen und Rotationen in der Regelung. Corke und Hutchinson [CH01] haben sich mit dem Problem beschäftigt, indem lediglich die translatorische und rotatorische Bewegung der z -Achse von den übrigen Freiheitsgraden entkoppelt wird.

Zusammenfassend kann beim Vergleich von IBVS und PBVS keine generelle Aussage über die Überlegenheit eines Prinzips getroffen werden. Bei IBVS ist das System sehr robust gegenüber Kalibrierungsfehlern, jedoch

ist gerade bei großen Regelfehlern die Trajektorie im kartesischen Raum unvorhersehbar und möglicherweise suboptimal. Bei PBVS kann die Trajektorie deutlich besser geplant werden, das Regelungsergebnis hängt aber direkt von der Qualität der Kalibrierung ab. Deng [DJSW02] und Janabi-Sharifi [JSDW11] haben Vergleiche zwischen IBVS und PBVS theoretisch und in Simulationen hinsichtlich der Kriterien Stabilität, Robustheit und Performance im kartesischen Raum als auch im Bildbereich erstellt. Es zeigt sich für beide Systeme asymptotische Stabilität und Robustheit gegenüber Kamerakalibrierung und Modellierungsfehlern. Insgesamt hat die Verwendung von Bildinformation im geschlossenen Regelkreis den Nachteil von Verzögerungen durch das Bildverarbeitungssystem und durch das Robotersystem. Diese können zwar präzisiert werden, wobei die Prädiktion jedoch sehr genaue Systemmodelle erfordert. Daher wird in dieser Arbeit das Dynamic Look-and-Move Prinzip verfolgt, um die Systemverzögerungen mithilfe der vorausschauenden Kamera berücksichtigen zu können.

1.3.2 Kraftregelung

Dieser Abschnitt widmet sich dem Stand der Technik auf dem Gebiet der Kraftregelung⁸⁾. Gerät das Werkzeug eines Roboters in Kontakt mit seiner Umgebung, ist eine Regelung der Kontaktkraft notwendig, um Beschädigungen an Werkzeug und Werkstück zu vermeiden. Häufig müssen prozessspezifische Kontaktkräfte hergestellt werden, wie beispielsweise beim robotergestützten Polieren von Oberflächen. Die Forschung im Bereich der Kraftregelung von Robotern begann bereits in den 1950ern. Whitney [Whi85] verfasste einen historischen Überblick über dieses Themengebiet.

Die zur Kraftregelung notwendige Kenntnis der Kontaktkraft erfolgt üblicherweise mittels Kraft-Momenten Sensorik (KMS). Dabei wird die Verformung eines definiert dehnbaren Elementes über Dehnungsmessstreifen erfasst und anschließend in die entsprechende Kraft umgerechnet [SSVO09]. Der Sensor wird dazu zwischen den Handflansch des Roboters und das mit der Umgebung in Kontakt tretende Werkzeug montiert (siehe Abbildung 1.6). Mit bekannten Abmessungen des Werkzeuges wird die Messung aus dem Sensorkoordinatensystem in den Kontaktpunkt zwischen Werkzeug und Umgebung transformiert, sodass die im Kontaktpunkt herrschende Kraft geregelt werden kann. Übersichten über die Kraftregelung von Ro-

⁸⁾Zur Vereinfachung des Sprachgebrauches bezieht nachfolgend die allgemeine Bezeichnung „Kraftregelung“ auch die „Momentenregelung“ mit ein.

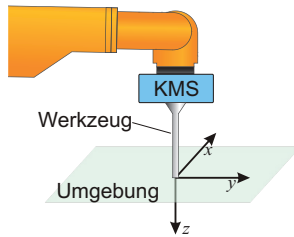


Abbildung 1.6: Montage des Kraft-Momenten-Sensors (KMS) zwischen Handflansch des Roboters und Werkzeug. Mithilfe der Werkzeugabmessungen wird die gemessene Kraft vom Sensorkoordinatensystem in den Kontaktpunkt zwischen Werkzeug und Umgebung transformiert.

botern sind z.B. von Yoshikawa [Yos00] und Zeng [ZH97] erstellt worden. Die Methoden kann man prinzipiell in implizite und explizite Kraftregelung unterteilen:

- **Implizite Kraftregelung:** Durch Krafrückführung wird das Verhalten des Roboters so beeinflusst, dass er sich wie eine mechanische Impedanz verhält, d.h. bei Vorgabe einer kartesischen Position wird das Verhalten eines mehrdimensionalen Feder-Masse-Dämpfer Systems erreicht. Man unterscheidet passive und aktive Impedanzregelung:
 - **Passive Impedanzregelung:** Der Roboter wird am Endeffektor mit Federn (und Dämpfern) ausgestattet, sodass sich für Fügeaufgaben der Kontaktpunkt relativ zum Endeffektor bewegen kann. In der Literatur wird diese Fügehilfe als Remote Center Compliance (RCC) beschrieben [KLY96, Lee05].
 - **Aktive Impedanzregelung:** Die Anfänge der Impedanzregelung lagen bei der reinen Steifigkeitsregelung. Dabei wird dem Roboter durch Krafrückführung das Verhalten einer programmierbaren Feder eingeprägt, indem das Verhältnis zwischen kartesischer Positionsdivergenz und den Gelenkmomenten unter Einbeziehung der Roboterkinematik beeinflusst wird [Sal80]. Die Impedanzregelung ist eine Verallgemeinerung der Steifigkeitsregelung und geht auf die Entwicklungen von Hogan zurück [Hog85]. Durch Beeinflussung der Relation zwischen Geschwindigkeit und Kraft (mechanische Impedanz) wird das Verhalten eines definierten Feder-Masse-Dämpfer Systems eingeprägt. Die aktive Impedanzregelung eignet sich z.B. zum manuellen Führen eines Roboters am Endeffektor.

- **Explizite Kraftregelung:** Im Gegensatz zur Impedanzregelung wird ein kartesischer Sollwert der Kraft explizit vorgegeben. Die Kontaktkraft wird in kartesischen Koordinaten geregelt. Dabei unterscheidet man kraftbasierte und positionsbasierte Verfahren. Volpe und Khosla vergleichen die beiden Verfahren zur expliziten Kraftregelung [VK92]:
 - **Kraftbasierte explizite Kraftregelung:** Unter Berücksichtigung der Kinematik des Roboters (mittels transponierter Jakobimatrix) wird aus der kartesischen Regeldifferenz der Kraft das Drehmoment auf Gelenkebene des Roboters berechnet. D.h., Stellgrößen werden direkt in Motordrehmomente umgerechnet.
 - **Positionsbasierte explizite Kraftregelung:** Über die Admittanz des Systems wird die Regeldifferenz der Kraft in ein Positionssignal umgerechnet [VK93]. Die Admittanz ist die Inverse der Impedanz - es wird somit die Steifigkeit, Masse und Dämpfung des Roboters in Kontakt mit der Umgebung berücksichtigt, sodass bei sehr weicher Umgebung schnelle Bewegungen und bei steifer Umgebung langsame Bewegungen erzeugt werden. Die positionsbasierte explizite Kraftregelung wird somit auch als Admittanzregelung bezeichnet, da dem Kraft-Regelfehler eine Endeffektorgeschwindigkeit zugeordnet wird [ZH97]. Diese Art der Kraftregelung ist bei den meisten kommerziellen Industrierobotern einsetzbar, da üblicherweise ein eingebauter Positionsregelkreis vorhanden ist, welcher über eine Schnittstelle zur Verarbeitung von kartesischen Positions- oder Geschwindigkeitsvorgaben verfügt. Häufig hat man jedoch keinen direkten Zugriff auf das Motordrehmoment, wie es für die kraftbasierte explizite Kraftregelung notwendig wäre.

Hybride Positions- und Kraftregelung

Da die meisten Anwendungen eine reine Kraft-/Momentenregelung in allen Achsrichtungen nicht erfordern bzw. nicht ermöglichen, muss Kraftregelung und Positionsregelung kombiniert eingesetzt werden. Ein gebräuchliches Beispiel ist die in Abbildung 1.7 dargestellte Bolzen-Loch Problematik. Dabei bedürfen einige Richtungen einer Kraftregelung und andere Richtungen einer Geschwindigkeitsvorgabe. Mason [Mas79, Mas81] begründet zur Definition dieser Richtungen eine Theorie, welche die Bewegungsrichtungen nach künstlichen und natürlichen Beschränkungen unterscheidet. Im Beispiel des Bolzen-Loch Problems ergeben sich die natürli-

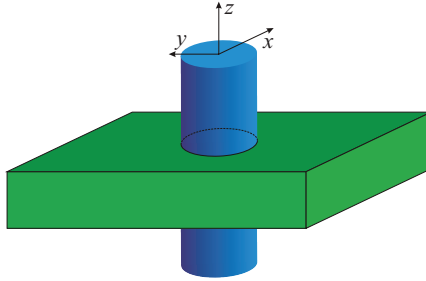


Abbildung 1.7: Bolzen-in-Loch Aufgabe (nach Mason [Mas81]). Der Bolzen wird in z -Richtung in das Loch eingeführt. Es wird von Reibungsfreiheit ausgegangen.

chen Beschränkungen folgendermaßen (es wird von einem reibungsfreien System ausgegangen):

$$\begin{aligned}
 v_x &= 0 & \omega_x &= 0 \\
 v_y &= 0 & \omega_y &= 0 \\
 F_z &= 0 & M_z &= 0
 \end{aligned} \tag{1.4}$$

Diese Beschränkungen besagen, dass sowohl in z -Richtung als auch um die z -Achse keine Kraft bzw. kein Moment ausgeübt werden kann ($F_z = 0, M_z = 0$). Zudem ist sowohl in x - und y -Richtung als auch um die x - und y -Achse keine Geschwindigkeit ungleich 0 möglich ($v_x = v_y = 0, \omega_x = \omega_y = 0$). Dazu ergeben sich komplementär die künstlichen Beschränkungen:

$$\begin{aligned}
 F_x &= 0 & M_x &= 0 \\
 F_y &= 0 & M_y &= 0 \\
 v_z &= k_1 & \omega_z &= k_2
 \end{aligned} \tag{1.5}$$

Die künstlichen Beschränkungen zeigen die Achsen an, in denen Bewegungen und Kräfte / Momente möglich sind und Sollwerte vorgegeben werden können. In dem Beispiel bedeutet dies, dass die Kräfte F_x und F_y als auch die Momente M_x und M_y mit einer Regelung zu 0 zu regeln sind. In der z -Richtung wird der Bolzen in das Loch mit einer gewünschten Geschwindigkeit k_1 geführt, während eine gewünschte rotatorische Geschwindigkeit k_2 erreicht wird.

Basierend auf der Formalisierung nach Mason haben Craig und Raibert die **hybride Positions- und Kraftregelung** eingeführt (siehe Abbil-

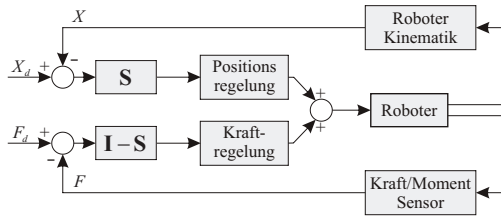


Abbildung 1.8: Hybride Positions- / Kraftregelung (nach Craig [CR79]). Mit den Auswahlmatrizen S und $I - S$ werden die Richtungen zur Positionsregelung bzw. Kraftregelung selektiert.

dung 1.8) [CR79, RC81]. Die Regelung wird in einem kartesischen System definiert und die künstlichen Beschränkungen in den Auswahlmatrizen S und $I - S$ berücksichtigt. D.h. die Auswahlmatrix S berücksichtigt die künstlichen Beschränkungen der Position und die dazu komplementäre Matrix $I - S$ berücksichtigt die künstlichen Beschränkungen der Kraft des Bolzen-Loch Problems (vgl. Gleichung 1.5) wie folgt:

$$S = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \quad I - S = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad (1.6)$$

Wie zuvor beschrieben, wird bei den meisten Systemen die positionsbasierte explizite Kraftregelung eingesetzt, da üblicherweise nur eine Schnittstelle zur kartesischen Positionsvorgabe verfügbar ist. Durch die unterlagerte Lageregelung, aber auch durch Filterung des Kraftsignales werden Verzögerungen im System eingeführt. Um Instabilität zu vermeiden, limitieren diese Verzögerungen jedoch stark die Grenzen der Reglerparameter einer diskreten Regelung [GTK10]. Durch diese Limitierung wird nur langsam auf Störgrößen reagiert, d.h., es besteht ein Konflikt zwischen der Unterdrückung von Messrauschen und der entstehenden Filterverzögerung [VK92]. Bei kraftgeregelter Konturverfolgung kann diese Limitierung zum Kontaktverlust führen, bzw. die Bewegungsgeschwindigkeit wird entlang der Kontur stark begrenzt [BS99]. Um den Einfluss der Grenze der positionsbasierten Kraftregelung zu reduzieren, wird in den nachfolgenden Abschnitten die parallele Kraft- und Bildregelung betrachtet.

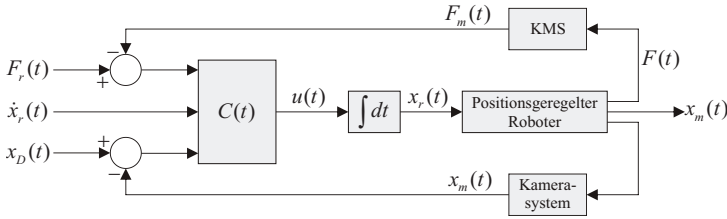


Abbildung 1.9: Kraft- und Bildinformationen in einem geschlossenen Regelkreis (nach Nelson [NMK96]). $C(t)$ definiert den Regler, um die Geschwindigkeitsvorgabe \dot{x}_r , die Vorgabe durch Kraftregelung F_r und die Vorgabe durch bildgestützte Regelung x_D zu berücksichtigen.

1.3.3 Parallele Kraft- und Bildregelung

Die in den vorigen Abschnitten beschriebenen Mechanismen zur Führung von Robotern durch Kraftregelung bzw. Bildregelung haben komplementäre Vorteile. Nelson et al. [NMK96, NMK95] beschreiben die Vorteile bildgestützter Regelung mit der globalen Information über ein relativ großes Gebiet, während der Vorteil des Kraftsensors bei hochpräziser aber lokal sehr begrenzter Information über die Kontaktsituation liegt. Nelson et al. stellen ein Framework vor, welches vergleichbar mit der hybriden Kraft- und Positionsregelung nach Craig [CR79, RC81] die Aufgabe in zueinander orthogonalen Richtungen definiert (siehe Abbildung 1.9). Das Framework nach Abbildung 1.9 unterscheidet drei Regelungsstrukturen, welche je nach Wahl des Reglers $C(t)$ aktiviert werden:

- **Traded Control:** In einer kartesischen Achsrichtung wird zwischen Geschwindigkeitsvorgabe, bildbasierter oder kraftbasierter Regelung während der Ausführung umgeschaltet. Dies eignet sich z.B. für eine Fügeapplikation: Zunächst wird die globale Sicht genutzt, um das Werkstück durch bildgestützte Regelung möglichst nahe an das Ziel zu führen, dann wird in der Kontaktrichtung von Visual Servoing auf Kraftregelung umgeschaltet, um das Bauteil letztendlich zu fügen. Dadurch kann eine hohe Geschwindigkeit der Annäherung erreicht werden, da lediglich in der Nähe des Ziels eine langsame Bewegungsgeschwindigkeit notwendig ist, um hohe Kräfte im Moment der Kontaktaufnahme zu vermeiden.

- **Hybrid Control:** Die Regelung von Kraft, Bild und Geschwindigkeit wird nur in zueinander orthogonalen Richtungen ausgeführt. Der Regler $C(t)$ enthält drei Selektionsmatrizen, deren Summe die Einheitsmatrix ergibt, d.h. in einer Achsrichtung wird entweder eine Geschwindigkeit vorgegeben oder kraftgeregelt oder Bildinformation verwendet. Bei einer Greifaufgabe wird z.B. die Ausrichtung des Werkzeuges in Bezug zum Ziel durch Visual Servoing realisiert, während eine Geschwindigkeit in der Annäherungsrichtung vorgegeben wird.
- **Shared Control:** Der Einsatz von Kraft- und Bildregelung in einer Achsrichtung wird gleichzeitig zugelassen. Wenn z.B. bei der bildgestützten Annäherung an eine Oberfläche die Position nicht sehr genau ermittelt werden kann, muss der Manipulator dennoch nachgiebig reagieren, auch in der durch Visual Servoing kommandierten Richtung. Morel et al. [MMB98] schlagen einen auf einem Impedanzregler basierenden Ansatz vor, wobei das Vision System die Referenztrajektorie in allen sechs Freiheitsgraden definiert, während der Roboter wie eine mechanische Impedanz bei unerwartetem Kontakt mit der Umgebung reagiert. Das Stabilitätsproblem wird durch Shared Control vereinfacht, da durch den Einsatz von Bildinformation eine Kraftregelung mit lediglich geringem Verstärkungsfaktor eingesetzt werden muss.

Die Kombination von Kraft und Bild erzeugt Verkopplungen der Messwerte: Eine durch Visual Servoing kommandierte Beschleunigung resultiert in der gleichen Achsrichtung in Trägheitskräften im Kraftsensor. Die Kraftmesswerte, welche nicht zur Kontaktkraft zählen, können den Kraftregelkreis destabilisieren. Wenn von einem idealen kartesischen Manipulator ausgegangen wird, vermeiden Hybrid- und Traded Control die Verkopplung, da Kraftregelung und Visual Servoing nicht gleichzeitig in der gleichen Richtung aktiv sind [NMK96]. Bei Shared Control ist jedoch eine Entkopplung dieser Richtungen durch Kraftkompensation notwendig, da bildgestützte Regelung und Kraftregelung in einer Achsrichtung gleichzeitig aktiv sind.

Kraftkompensation

Um in dynamischen Situationen die Destabilisierung des Kraftreglers durch Trägheitskräfte zu vermeiden, berücksichtigt Nelson die zweifache Ableitung der kartesischen Position des Roboters. Dabei ignoriert er den Ausgang des Kraftreglers solange eine deutliche Beschleunigung durch Vi-

sual Servoing kommandiert wird. Die direkte Berechnung und Kompensation der Trägheitskräfte wird aufgrund der verrauschten Beschleunigungssignale aus zweifacher Ableitung nicht durchgeführt. Ist die Trajektorie der Roboterbewegung bekannt, kann mithilfe dieses Wissens die Trägheitskraft berechnet und von der Kraftmessung subtrahiert werden. In sensorgeführten Aufgaben ist diese Trajektorie jedoch nicht von vornherein bekannt. Winkler [WS07] stellt die Kraft- und Momentenkompensation mit 12DOF Kraft-Momenten-Beschleunigungssensoren vor. Basierend auf der Lastidentifikation und der Messung der Beschleunigung mit integrierten Beschleunigungssensoren wird die Kompensation der Trägheitskräfte durchgeführt. García [GROJ04] fusioniert Beschleunigungssensoren mit der Kraftsensorik in einem Kalman-Beobachter, um Kontaktkräfte zu ermitteln. In einer Erweiterung wird zusätzlich die Roboterdynamik mit berücksichtigt, um das Verhalten des Beobachters verbessert auslegen zu können [GROJ05].

Die Messung der Beschleunigung kann auf unterschiedliche Weisen realisiert werden. Wu [WMS04] entwirft eine monolithische seismische Masse und misst über Dehnungsmessstreifen die Beschleunigung in allen 6 Achsrichtungen. Weiterhin existieren kapazitive [PR98], piezoresistive [ADTS07] und piezoelektrische [KES01] Verfahren. Heutige Entwicklungen dieser Sensoren basieren jedoch auf der MEMS Technologie [BRB⁺06]. Kröger [KKW08] gibt einen Überblick über Beschleunigungsmessungen. Rotatorische Beschleunigungen werden aus einer Anordnung von Beschleunigungssensoren auf einem Messradius um die Drehachse berechnet. Diese Berechnung rotatorischer Beschleunigung unterliegt starken Störungen durch Empfindlichkeit auf Vibrationen und Übersprechen der Sensorachsen, sodass kein verwertbares Signal extrahiert werden kann. Auch die Ermittlung der rotatorischen Beschleunigung durch zweifache Ableitung der Roboterpose erzeugt starkes Rauschen und unterliegt Störungen aus der Transition der Bewegung durch die gesamte kinematische Kette des Roboters. Kröger verwendet zur Berechnung der rotatorischen Beschleunigung einen Kalman-Filter zur Glättung der Roboterpose.

Viele Aufgaben in der Robotik werden mit der Voraussetzung steifer Werkstücke und Werkzeuge definiert. In diesen Fällen genügt die Entkopplung zwischen Kraftregelung und bildgestützter Regelung durch Kraftkompensation. Bei nachgiebigen Objekten bestehen aber nicht nur Kopplungen über die Trägheit, sondern der Kontakt mit der Umgebung verformt die Umgebung, sodass visuell erfasste Merkmale im Bild entsprechend der Verformung verschoben werden. Luo und Nelson [LN00] modellieren Ob-

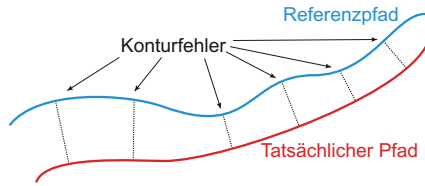


Abbildung 1.10: Der Konturfehler ist die Abweichung des tatsächlichen Pfades vom Referenzpfad (nach [CL07])

jektverformung durch eine Finite Element Methode (FEM), um vorherzusagen, welche Verformung durch Kontaktkraft hervorgerufen wird. Baeten und Schutter [BS99] kompensieren bei einer Konturverfolgung die Position der Kamera aufgrund der Verformung des Werkzeuges durch die aktuelle Kontaktkraft und verbessern somit das Ergebnis der Kamerainformation.

1.3.4 Konturverfolgung

Eine Anwendung von Kraftregelung und/oder Visual Servoing kann bei der Konturverfolgung gefunden werden. Dabei wird die Position (und Orientierung) des robotergeführten Werkzeuges an eine Werkstückkontur angepasst. Diese Aufgabe kann entweder als reines Bildverarbeitungsproblem, als reine Kraftregelung oder auch als eine Kombination von beidem realisiert werden. Das Ziel ist die Reduktion des Konturfehlers zwischen der gewünschten und der tatsächlichen Trajektorie des Roboters (siehe Abbildung 1.10).

De Schutter und Bruyninckx [DSB88, BDSB03] formalisieren die hybride Kraft-/Positionsregelung bzw. den Shared Control Ansatz im Task Frame Formalism (TFF). In dieser Formalisierung wird zwischen verschiedenen Formen von Shared Control unterschieden. Die Aufgabe im Task Frame (Koordinatensystem der Aufgabe bzw. des Werkzeuges) ist in die 6 kartesischen Freiheitsgrade unterteilt. Für jede der 6 Achsrichtungen wird jeweils die Bewegung durch eine (oder eine Kombination) der folgenden Vorgaben definiert:

- **Kraft/Moment:** Eine explizite Kontaktkraft bzw. -moment zwischen Werkzeug und Werkstück wird aufgebaut.

- **Geschwindigkeit:** Die Bewegung wird durch einen Geschwindigkeits-sollwert definiert, sofern keine natürliche Beschränkung in der entsprechenden Achsrichtung besteht.
- **Tracking:** Aus einer tangentialen Bewegung entlang einer Kontur und der Positionskorrektur durch eine Kraftregelung wird die Orientierung berechnet, die notwendig ist, um den Task Frame z.B. orthogonal zur Kontur zu halten.
- **Visual Servoing:** Bewegung entlang einer Achse des Task Frames in Abhängigkeit von Bildinformationen.
- **Feedforward:** Durch online-Messung oder modellbasiertes Wissen wird eine Vorsteuerung entweder einer Tracking-Richtung, einer kraftgeregelten Richtung oder einer bildgestützten Regelung überlagert, um die jeweilige Regelabweichung zu reduzieren. Die Vorsteuerung durch Kamerainformation ist z.B. bei einer rein kraftgestützten Konturverfolgung nicht zwingend notwendig, aber es erhöht dennoch die Qualität bzw. die Geschwindigkeit der Bewegung, da die notwendige Korrektur durch den Kraftregler reduziert wird. Soll z.B. eine „Hügellandschaft“ durch Kraftregelung verfolgt werden, kann durch Vorsteuerung in Richtung der Kraftregelung die Abweichung der Kontaktkraft vom Sollwert drastisch reduziert werden.

Methoden zur Konturverfolgung in der Literatur unterscheiden sich sowohl im Kamerasystem und in der Reglerstruktur als auch in den getroffenen Einschränkungen. Nachfolgend werden einige Ansätze der Konturverfolgung vorgestellt:

- **Konturverfolgung auf der Basis von Visual Servoing.** Aus der Bildaufnahme wird eine Verschiebung des Manipulators berechnet, um diesen an einer Kontur auszurichten. Coste-Manière [CMCK93] ermittelt aus einem Mono-Kamerasystem die Objektkontur. Die y -Achse des Werkzeugkoordinatensystems wird tangential an der Kontur ausgerichtet, indem bei konstanter Geschwindigkeit in y -Richtung bestimmte Bildmerkmale an einer konstanten Stelle im Bild gehalten werden [CMCK95]. Die Kamera ist parallel zur Bildebene, sodass dieses System auf planare Aufgaben beschränkt ist. Jeon [JABH07] wendet die Tangentenapproximation am hell-dunkel Übergang zwischen Objekt und Umgebung zur Führung eines mobilen Roboters an. Auch dieser Ansatz ist auf planare Aufgaben mit kontrastreichen Konturverläufen beschränkt. Gangloff [GM02] verfolgt 3D-Profile in allen Freiheitsgraden mit einer Kamera. Durch die Forderung der Sichtbarkeit von drei par-

allelen Profilkanten mit bekannten und konstanten Abständen wird aus der Bildaufnahme die Kameraposition in Bezug zum Profil ermittelt. Dieses System ist auf nicht gewundene Profile mit bekanntem Querschnitt beschränkt.

- **Modellbasierte Konturverfolgung.** Demey [DS95] verbindet online generierte Messwerte mit offline generierter Modellinformation zur planaren Konturverfolgung mit Robotern. Die Trajektorie wird z.B. aus einem CAD Modell oder einer Kameraaufnahme vorgegeben, sodass zu jedem Punkt eine gewünschte Position, Geschwindigkeit und Kraft definiert ist. Über die Kraftregelung wird bei Bewegung des Roboters die Krümmung des Merkmales erfasst (Tracking) und mit einem Matching Algorithmus einer Position entlang des Modells zugeordnet. Dadurch kann ein Feedforward Signal generiert werden, um die reine Kraftregelung zur Konturverfolgung zu „entlasten“.
- **Konturverfolgung mit prädiktivem Dynamic Look-and-Move.** Die Bildinformation wird nicht in einem geschlossenen Regelkreis verwendet, jedoch wird ein Referenzpfad entlang der Kontur benötigt [LWH98]. Die Abweichung vom Referenzpfad wird mit einer Monokamera nicht nur am TCP sondern auch vorausschauend ermittelt. Dadurch wird nicht nur die aktuelle Abweichung am TCP berücksichtigt, sondern über eine Gewichtung auch zukünftige Sollwerte. Das System ist auf planare Anwendungen beschränkt. Luo [LC04] verwendet ein vorausschauendes Lasertriangulationssystem und berücksichtigt den Kameravorlauf durch Verzögerung der ermittelten Stellwerte um eine konstante Zeit, basierend auf der Annahme einer konstanten Geschwindigkeit und konstanter Vorlaufstrecke des Sensors.
- **Kraft- und bildgestützte Konturverfolgung.** Baeten et al. [BS99, BDS01, BS02] beschreiben planare Konturverfolgung mit Bild- und Kraftsensorik an unbekannten Objekten. Eine spezifizierte tangentielle Geschwindigkeit entlang der Kontur und eine spezifizierte Kontaktkraft in Normalenrichtung werden vorgegeben. Die Aufgabe wird im Task Frame Formalism (TFF) definiert. Es wird eine Krafrichtung, geschwindigkeitsgeregelter Richtung, Anpassung der Orientierung aus Kraft-Momenten Messung und bildgestützte Vorsteuerung berücksichtigt. Die Vorsteuerung unterstützt die Anpassung der Orientierung, indem die Ausrichtung der Kontur im Vorlauf ermittelt wird. Durch die Vorausschau werden Verzögerungen des geschlossenen Kraftregelkreises reduziert. Zudem können Ecken vorzeitig erkannt werden und die Bewegungsgeschwindigkeit entsprechend reduziert werden, um das

Werkzeug in einer Ecke zu drehen. Die vorgestellten Experimente sind aufgrund der Kameraanordnung auf planare Aufgaben beschränkt.

Grundsätzlich existieren viele erfolgreiche Forschungsarbeiten auf dem Gebiet der Konturverfolgung. Einige der Arbeiten unterliegen jedoch starken Einschränkungen, die die Anwendbarkeit stark spezialisieren. Zu den Einschränkungen zählt sehr häufig die Begrenzung auf planare Aufgaben, da unkalibrierte Mono-Kamerasysteme bevorzugt werden. Weitere Einschränkungen werden an die Objekte gestellt, sodass entweder sehr hoher Kontrast zwischen Objekt und Umgebung vorhanden sein muss, oder bestimmte Objektgeometrien bekannt sein müssen. Darüber hinaus sind Ansätze des Visual Servoing nur solange möglich, wie der zu regelnde Kontaktpunkt während der Applikation sichtbar ist. In realen Anwendungen besteht jedoch häufig eine Verdeckung des Kontaktpunktes durch das Werkzeug selbst. Weitere Einschränkungen kommen aus der Algorithmetik, sodass von konstanten Vorlaufstrecken und konstanten Bewegungsgeschwindigkeiten ausgegangen wird, um die im Vorlauf generierten Messwerte der aktuellen Werkzeugposition zuzuordnen.

In dieser Arbeit wird ein Lasertriangulationssystem angewendet, sodass auch Konturen an kontrastarmen Objekten erkannt werden. Die Messung erfolgt im Vorlauf, sodass keine Verdeckungen auftreten und es werden keine Einschränkungen an Freiheitsgrade oder Geschwindigkeitsprofile gemacht, da die Messwerte dreidimensional mit dem kalibrierten Kamerasystem aufgezeichnet werden. Verzögerungen durch die Verwendung der Bildinformation im geschlossenen Regelkreis werden durch die Vorlaufmessung umgangen, und die Verzögerung des Robotersystems prädiktiv in die Trajektorienplanung mit einbezogen. Die bildgestützte Konturverfolgung in diesem System wird entsprechend der vorangegangenen Klassifizierung als prädiktives Dynamic-Look-and-Move realisiert.

1.4 Systemaufbau

In dieser Arbeit wird ein breites Spektrum an Experimenten vorgestellt. Dazu gehören Experimente zur Kraftkompensation, Verbesserung der kartesischen Positionsermittlung des Roboters und Konturverfolgung mit und ohne Kontakt an stetigen und nicht-stetigen Konturen auf der Basis von Bild-, Kraft- und Beschleunigungssensorik. Dazu wurde im Zentrum für

Robotik an der Hochschule Darmstadt ein umfangreiches Experimentalsystem aus folgenden Komponenten aufgebaut:

1. Robotersystem KUKA KR60-2
2. Kraft-Momenten-Sensor (KMS)
3. Kamerasystem
4. Beschleunigungssensor
5. Computersystem

Nachfolgend werden die Komponenten beschrieben. Eine Übersicht der Komponenten und deren Verbindung ist in Abbildung 1.11 dargestellt.

Robotersystem KUKA KR60-2

Für die Experimente wird ein 6-Achs Industrieroboter KUKA KR60-2 eingesetzt. Die Steuerung des Roboters erfolgt durch die KUKA KRC-2 Steuerung. Bei einer Traglast von $60kg$ wird eine maximale kartesische Geschwindigkeit von $v_{max} = 2 \frac{m}{s}$ und eine maximale Beschleunigung von $a_{max} = 10 \frac{m}{s^2}$ erreicht. Die Wiederholgenauigkeit wird mit $0,2mm$ angegeben.

Das *Robot Sensor Interface (RSI)* stellt eine Schnittstelle zur Verfügung, um mit einem externen PC via Ethernet zu kommunizieren. Dadurch kann die aktuelle Pose des Roboters im Interpolationstakt T_{IPO} auf kartesischer Ebene und auf Gelenkebene abgefragt werden. Ebenso kann die Sollpose auf kartesischer Ebene und auf Gelenkebene vorgegeben werden. Der Interpolationstakt der KRC-2 Steuerung liegt bei $T_{IPO} = 12ms$. Ab 2011 stand ein experimenteller Softwarestand für die Steuerung zur Verfügung, sodass der Interpolationstakt auf $T_{IPO} = 4ms$ gesenkt werden konnte⁹⁾.

Abbildung 1.12(a) und (b) zeigen die Messung der Istposition (rot) bei sprungförmiger Änderung der Sollposition via RSI (blau) um $\Delta x = 1mm$. Eine Totzeit des Systems von $T_t = T_{IPO} = 12ms$ ist erkennbar. Die Geschwindigkeit der Umsetzung wird mit einer systeminternen Filtervariable beeinflusst. Es ergibt sich die folgende Anregelzeit T_{An} aus der Totzeit und der Filterverzögerung T_{Filter} :

$$T_{An} = T_{IPO} + T_{Filter} \quad (1.7)$$

⁹⁾Aus dem Systemwechsel folgt, dass nicht alle vorgestellten Experimente mit der gleichen Abtastrate durchgeführt wurden.

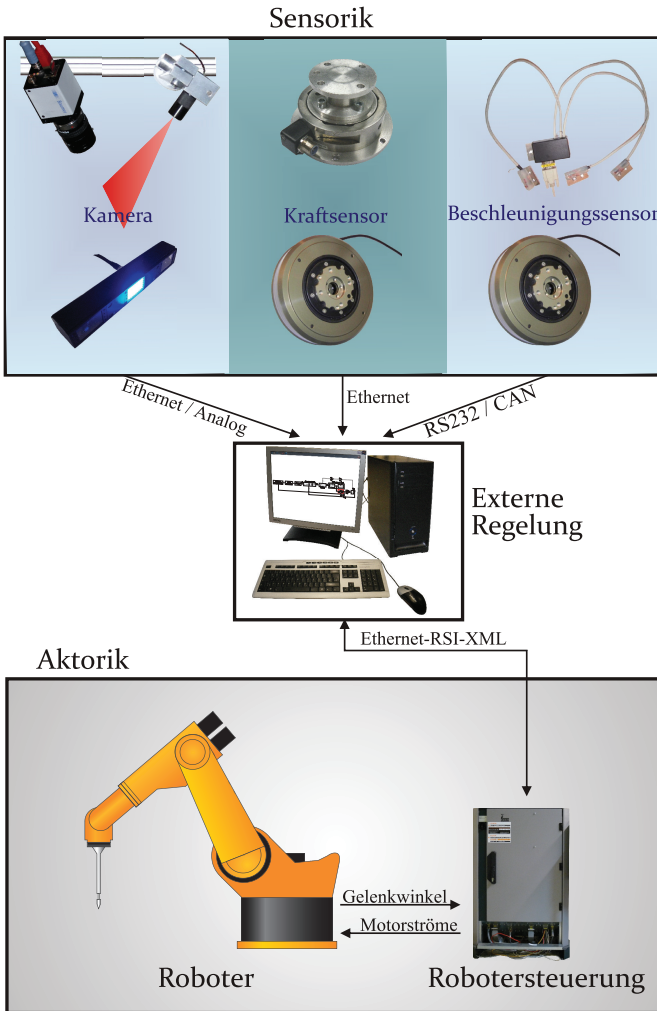
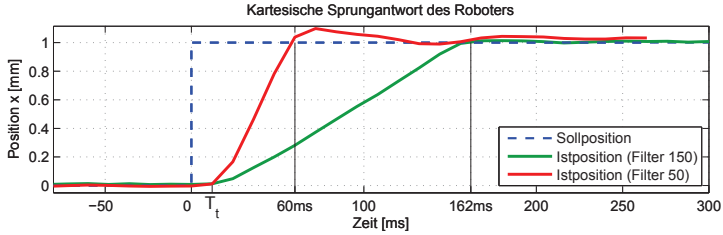
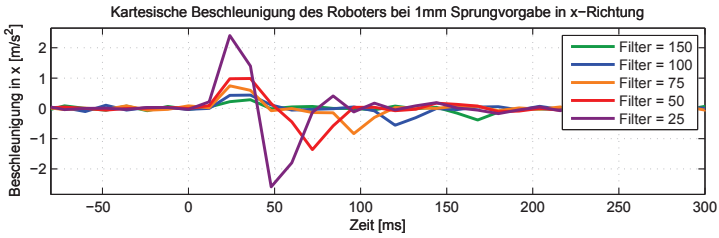


Abbildung 1.11: Aufbau des Experimentalsystems an der Hochschule Darmstadt. Der Regelungs-PC verarbeitet die Daten der angeschlossenen Sensorik und des Industrieroboters KUKA KR60-2.



(a) Sprungantwort mit Filtereinstellungen von $T_{filter} = 150ms$ und $T_{filter} = 50ms$. Es ergibt sich eine Anregelzeit von $T_{An} = 162ms$ bzw. $T_{An} = 60ms$. Bei schnellerer Anregelzeit ergibt sich ein deutliches Überschwingen.



(b) Beschleunigung (zweite Ableitung der Istposition) bei Sprungantwort mit unterschiedlicher Filtervariable.

Abbildung 1.12: Anregelzeit und Beschleunigung des Robotersystems bei sprungförmiger Änderung der Sollposition um $\Delta x = 1mm$ bei unterschiedlicher Einstellung der systeminternen Filtervariable.

Die Wahl einer sehr kleinen Filtervariable lässt deutlich höhere Dynamik in der Roboterbewegung zu, jedoch steigen damit die Beschleunigungen im System. Abbildung 1.12(b) zeigt die Beschleunigungen bei einer sprungförmigen Änderung der Position um $\Delta x = 1mm$ bei unterschiedlichen Filtervariablen.

Kraft-Momenten-Sensor

Zur Messung von Kräften und Momenten kommen zwei unterschiedliche Sensoren zum Einsatz:

- Nachgiebiger Kraft-Momenten-Beschleunigungs-Sensor FTAC 50-80 der Firma Schunk. Elastomere verbinden das Gehäuse mit dem werkzeugseitigen Flansch; die Auslenkung des Flansches wird optisch mittels PSD-Dioden (Position Sensitive Device) ermittelt. Durch die herstellerseitige Kalibrierung wird einer Auslenkung eine Kraft zugeordnet. Der werkzeugseitige Flansch kann sich somit um $\pm 1,4\text{mm}$ gegenüber dem Gehäuse bewegen. Der Messbereich der Kräfte und Momente ist in Tabelle 1.1 dargestellt. Gehäuseseitig sind zusätzlich drei 2-Achs Beschleunigungssensoren mit einem Messbereich von $\pm 2g$ verbaut. Die Anordnung der Sensoren ermöglicht die Messung von translatorischen und rotatorischen Beschleunigungen. Aufgrund der Messung der Kräfte und Momente in allen sechs Raumrichtungen und der Messung der translatorischen und rotatorischen Beschleunigung in allen sechs Raumrichtungen wird dieser Sensor nachfolgend mit „12DOF Sensor“ bezeichnet. Die Kommunikation erfolgt über RS232 oder CAN-Bus bei einer Abtastfrequenz von bis zu 1kHz .
- Steifer Kraft-Momenten-Sensor DELTA-SI-660-60 der Firma ATI Industrial Automation (vertrieben durch Schunk). Durch die steife Konstruktion des Sensors ergibt sich zwischen Gehäuse und werkzeugseitigem Flansch nur eine minimale Auslenkung in Abhängigkeit der Kraft, welche durch Dehnungsmessstreifen erfasst wird. Der Messbereich der Kräfte und Momente ist in Tabelle 1.1 dargestellt. Die Kommunikation erfolgt über Ethernet bei einer maximalen Abtastfrequenz von 7kHz . Aufgrund der 6 Freiheitsgrade wird dieser Sensor nachfolgend mit „6DOF Sensor“ bezeichnet.

Kamerasystem

Zur dreidimensionalen Erfassung von Konturen stehen zwei Lichtschnittsensoren zur Verfügung. Beide Sensoren werden am Roboter montiert.

- Industrieller Triangulationssensor RGS300 der Firma ISRA. Durch eine Optik werden sechs Lichtstreifen auf das Werkstück projiziert. Eine 2D CCD-Kamera erfasst die Projektion. Das analoge Kamerasignal wird mittels Framegrabber im PC digitalisiert. Durch eine

Tabelle 1.1: Messbereiche der Kraft-Momenten-Sensoren (Überlastbereiche sind in Klammern angegeben)

| Bezeichnung | Kräfte [N] | | | Momente [Nm] | | |
|---------------------|-------------------|---------------|-----------------|---------------------|-------------|-------------|
| | F_x | F_y | F_z | M_x | M_y | M_z |
| FTAC 50-80 | 300 (400) | 300 (400) | 300 (350) | 7 (14) | 7 (14) | 15 (25) |
| DELTA-SI -660-60 | 660 (3500) | 660 (3500) | 1980 (11900) | 60 (230) | 60 (230) | 60 (420) |

Kalibrierung ist sowohl der Zusammenhang zwischen Beleuchtung und Kamera als auch der Zusammenhang zwischen Kamerasystem und Roboter bekannt, sodass dreidimensionale Messwerte innerhalb der Lichtstreifen auch bei bewegtem Roboter ermittelt werden können.

- Eigenentwickelter Laser-Triganulationssensor. Über eine Optik wird eine einzelne Laserlinie auf das Werkstück projiziert. Eine CCD-Kamera TXG06 der Firma Baumer mit Gigabit Ethernet Schnittstelle erfasst die Projektion auf dem Werkstück. Die Auswertung erfolgt nach dem gleichen Messprinzip wie bei dem industriellen Sensor.

Beschleunigungssensor

Zur Messung von Beschleunigungen stehen zwei unterschiedliche Sensoren zur Verfügung:

- Nachgiebiger Kraft-Momenten-Beschleunigungs-Sensor der Firma Schunk (Beschreibung siehe „Kraft-Momenten-Sensor“).
- Drei 2-Achs Beschleunigungssensoren ADXL 202 der Firma Analog Devices. Der analoge Messwert mit einem Messbereich von $\pm 2g$ wird über einen ATMEL Mikroprozessor ATMega32 mit 10Bit Analog Digital Wandlung verarbeitet. Die Kommunikation erfolgt über RS232 bei einer Abtastrate von $1kHz$. Die drei einzelnen Sensoren sind über $50cm$ lange Kabel mit dem Mikroprozessor verbunden. Diese Bauweise erlaubt es, die drei Sensoren beliebig auf dem Roboterflansch bzw. Werkzeug anzuordnen.

Computersystem

Die Regelung wird auf einem Standard-PC mit Microsoft Windows XP Betriebssystem realisiert. Es steht ein PC mit Intel Core2Duo Prozessor mit $3,0GHz$ und $4GB$ Arbeitsspeicher zu Verfügung. Es besteht eine Verbindung zu den verschiedenen Sensoren als auch zum Roboter. Dieses System stellt kein Echtzeitsystem zur Verfügung. Die Robotersteuerung läuft jedoch auf einem Echtzeit Betriebssystem. Über die Kommunikation mit dem Roboter wird die PC-seitige Regelung im $12ms$ bzw. $4ms$ Takt getriggert, sodass von einem Quasi-Echtzeit System ausgegangen werden kann.

2 Ermittlung der kartesischen Roboterposition

Für eine positionsbasierte Regelung, wie z.B. *Position-Based Visual Servoing* (PBVS), ist die exakte Kenntnis der aktuellen Pose des TCP (Tool-Center-Point) von hoher Bedeutung, da eine Trajektorie in kartesischen Koordinaten vorgegeben wird. In diesem Kapitel werden Messverfahren auf der Basis der Gelenkwinkel des Roboters und auf Basis von Beschleunigungssensorik miteinander verglichen (siehe Abbildung 2.1).

Die kartesische Position und Orientierung des TCP wird aus der Vorwärtstransformation des Roboters berechnet. Mittels z.B. der Denavit-Hartenberg Konvention [SSVO09] kann in Abhängigkeit der Abmessung-

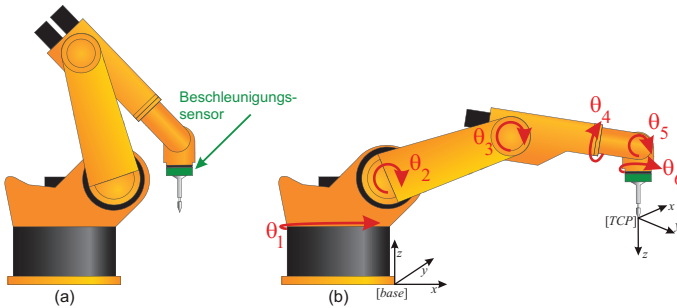


Abbildung 2.1: Die Position und Orientierung des TCP wird aus den Gelenkwinkeln $\theta_i, i = 1, \dots, 6$ mittels der Vorwärtstransformation berechnet. Die Massenträgheit bezüglich der Gelenkachsen beeinflusst die Qualität der Messung und hängt von der aktuellen Stellung des Roboters ab: (a) niedrige Trägheit bezüglich der Gelenke 1-3, (b) hohe Trägheit bezüglich der Gelenke 1-3. Ein Beschleunigungssensor ist zusätzlich am Handflansch befestigt, um die Dynamik tatsächlicher Bewegungen zu erfassen.

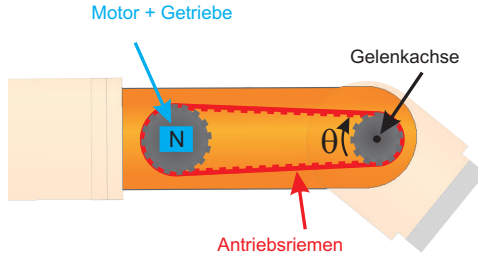


Abbildung 2.2: Darstellung eines Antriebsstranges bestehend aus Getriebe mit Übersetzungsverhältnis N , Motor und Antriebsriemen.

en und Anordnung der sechs Arnteile und den aktuellen Gelenkwinkeln $\theta_i, i = 1, \dots, 6$ die homogene Transformation \mathbf{T}_{base}^{TCP} zwischen dem Basiskoordinatensystem $K_{[base]}$ und dem Werkzeugkoordinatensystem $K_{[TCP]}$ berechnet werden.

Die Gelenkwinkel θ_i werden nicht direkt an der Gelenkachse (Abtrieb) gemessen, sondern an der entsprechenden Motorwelle des zugehörigen Antriebs. Die über Resolver erfassten Motorwinkel $\Theta_i, i = 1, \dots, 6$ werden mit dem Übersetzungsverhältnis $N_i, i = 1, \dots, 6$ der Getriebe auf Gelenkwinkel umgerechnet. Der prinzipielle Aufbau eines Antriebsstranges ist in Abbildung 2.2 dargestellt. Die Messung des Motorwinkels erhöht die Auflösung des Messsystems um die jeweiligen Getriebefaktoren. Im stationären Zustand kann der Gelenkwinkel θ_i mit hoher Genauigkeit ermittelt werden. Bei Bewegungen mit hoher Beschleunigung ergibt sich jedoch durch die endliche Steifigkeit der Armteile bzw. des Antriebsstranges eine Verzögerung zwischen Antrieb und Abtrieb. Dadurch entstehen bei der Berechnung der kartesischen Position des TCP Abweichungen zur tatsächlichen Position. Die Verzögerung zwischen Antrieb und Abtrieb hängt von der Massenträgheit bezüglich der Gelenkachsen und der aktuellen Beschleunigung ab. Die Trägheit selbst hängt von der Nutzlast sowie von der aktuellen Stellung des Roboters ab (vgl. Abbildung 2.1). Die Modellierung der Verzögerung zwischen Antrieb und Abtrieb in Abhängigkeit der aktuellen Beschleunigungen und Stellungen des Roboters wäre nur eingeschränkt genau, da dazu ein exaktes Modell des Gesamtsystems notwendig ist.

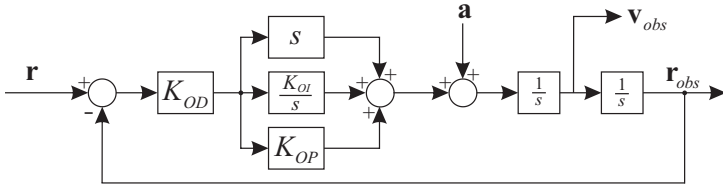


Abbildung 2.3: Modifizierter Rigid-Body Luenberger Observer (RBO) (nach [EL00]) zur Beobachtung der Roboterposition auf Basis der Position \mathbf{r} aus der Vorwärtstransformation und der Beschleunigung \mathbf{a} .

Durch den Einsatz von zusätzlicher Sensorik direkt am TCP des Roboters kann die Genauigkeit des kartesischen Positionssignales jedoch erhöht werden. In diesem Abschnitt wird ein Beobachter vorgestellt, welcher das Signal eines Beschleunigungssensors und die kartesische Position aus der Vorwärtstransformation auf Basis der Gelenkwinkel berücksichtigt. Der Beschleunigungssensor ist am Roboterhandflansch befestigt (siehe Abbildung 2.1(a)). Da die ermittelte Beschleunigung die tatsächliche Bewegung des TCP widerspiegelt, kann mithilfe dieser Messung die Position des TCP bei hohen Beschleunigungen genauer ermittelt werden.

Im Bereich der Regelung von Servosystemen schlagen Ellis und Lorenz [EL00] den *Rigid-Body Luenberger Observer* (RBO) vor. Die von Ellis und Lorenz vorgeschlagene Struktur verwendet den Motorwinkel als Eingang und das vom Servoverstärker kommandierte Drehmoment als Vorsteuerung, um die Winkelgeschwindigkeit des Servosystems zu ermitteln (das Motordrehmoment entspricht in Abhängigkeit des Massenträgheitsmoments der Winkelbeschleunigung). Dieser Ansatz wird auf die kartesische Ebene übertragen, indem das kartesische Positionssignal und ein Beschleunigungssignal verwendet werden. In Abbildung 2.3 ist die modifizierte Struktur dargestellt. Als Eingangssignale werden verwendet:

- Kartesische Position $\mathbf{r} = (r_x, r_y, r_z)^T$ des TCP aus der Vorwärtstransformation in Abhängigkeit der Gelenkwinkel $\theta_i, i = 1, \dots, 6$. Die Berechnung von \mathbf{r} erfolgt in der Robotersteuerung.
- Beschleunigung $\mathbf{a} = (a_x, a_y, a_z)^T$ des TCP durch Messung mit einem 3-Achs Beschleunigungssensor am Handflansch des Roboters.

Die von der Robotersteuerung berechnete Position \mathbf{r} dient als Eingang und die Beschleunigung \mathbf{a} als Vorsteuerung. Die Eingangsdifferenz zwi-

schen Position \mathbf{r} und beobachteter Position $\mathbf{r}_{obs} = (r_{obs,x}, r_{obs,y}, r_{obs,z})^T$ ist der Beobachtungsfehler. Dieser Fehler wird durch einen PID Regler zu Null geregelt. Ausgehend von der beobachteten Geschwindigkeit \mathbf{v}_{obs} kann die Übertragungsfunktion des in Abbildung 2.3 dargestellten Beobachters folgendermaßen dargestellt werden:

$$\mathbf{v}_{obs} = \underbrace{\frac{\mathbf{a}(s)}{s}}_{\mathbf{v}_{int}} \cdot \frac{s^3}{s^3 + K_{OD}(s^2 + K_{OP}s + K_{OI})} + \underbrace{\mathbf{r}(s) \cdot s}_{\mathbf{v}_{diff}} \frac{K_{OD}(s^2 + K_{OP}s + K_{OI})}{s^3 + K_{OD}(s^2 + K_{OP}s + K_{OI})} \quad (2.1)$$

Nach [Unb07] kann der Nenner in die Form des Nenners eines Butterworthfilters mit der Grenzfrequenz ω_g überführt werden:

$$s^3 + K_{OD}s^2 + K_{OD}K_{OP}s + K_{OD}K_{OI} = s^3 + 2\omega_g s^2 + 2\omega_g^2 s + \omega_g^3 \quad (2.2)$$

Aus der Vorgabe einer Grenzfrequenz $\omega_g = 2\pi f_g$ ergeben sich für die Filterkoeffizienten die folgenden Werte:

$$K_{OD} = 2\omega_g \quad (2.3)$$

$$K_{OD}K_{OP} = 2\omega_g^2 \rightarrow K_{OP} = \omega_g \quad (2.4)$$

$$K_{OD}K_{OI} = \omega_g^3 \rightarrow K_{OI} = \frac{\omega_g^2}{2} \quad (2.5)$$

Die Gleichungen 2.1 bis 2.5 können folgendermaßen als Filter interpretiert werden:

- Durch die Integration des Signales \mathbf{a} aus dem Beschleunigungssensor ergibt sich die Geschwindigkeit $\mathbf{v}_{int} = \frac{\mathbf{a}(s)}{s}$, welche durch Gravitation und Drift stark beeinflusst ist. Durch Angabe der Grenzfrequenz ω_g wird diese Geschwindigkeit mit einem **Hochpass gefiltert**, sodass der niederfrequenten Anteil herausgefiltert wird. Das Signal wird somit von Drift und stationären Anteilen (Gravitation) befreit.
- Durch die Ableitung der Roboterposition \mathbf{r} ergibt sich die Geschwindigkeit $\mathbf{v}_{diff} = \mathbf{r}(s) \cdot s$, die durch Quantisierungsrauschen beeinflusst wird. Zudem sind die zuvor beschriebenen Verzögerungen enthalten, die sich bei Bewegungen mit starker Beschleunigung ergeben. Durch Angabe der Grenzfrequenz ω_g wird die abgeleitete Geschwindigkeit mit einem **Tiefpass gefiltert**. Dadurch werden die hochfrequenten

Anteile aus dem Signal entfernt, und es verbleiben lediglich die hochgenauen stationären bzw. niederfrequenten Anteile im Signal.

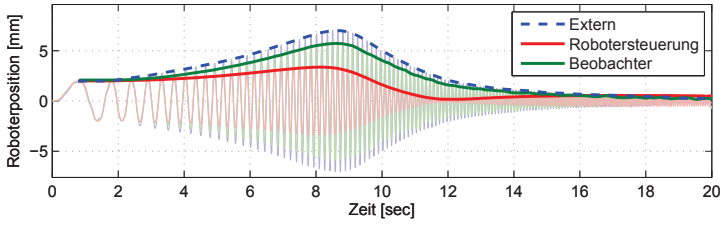
- Durch die **Addition** der beiden gefilterten Signale werden die Vorteile beider Messsysteme verbunden: Der hochgenaue stationäre Anteil der Roboterposition aus der Robotersteuerung und die hohe Dynamik des Beschleunigungssensors am Handflansch.

Abbildung 2.4 zeigt die beobachtete Position bei hoher Massenträgheit (vgl. Abbildung 2.1). Die Grenzfrequenz des Beobachters wurde für alle Messungen auf $\omega_g = 2\pi \cdot 2Hz$ gesetzt. Die Sollposition des Roboters wird in y -Richtung des Basiskoordinatensystems als Chirpsignal¹⁾ mit der Frequenz $f(t) = 0,75 \cdot t$ und einer Amplitude von $2mm$ vorgegeben. Es werden nur Frequenzen bis $15Hz$ vorgegeben, da höhere Frequenzen Schwingungen im System erzeugen, die den Roboter beschädigen können. Die tatsächliche Position des TCP wird mit einem externen Kamerasystem ermittelt und bildet somit das Referenzsignal für die Messung (blau gestrichelt). Die Soll-Vorgabe der Amplitude von $2mm$ wird nicht eingehalten, sodass durch Aufschwingen des Systems die Amplitude auf das 3-fache der Vorgabe steigt. Jedoch soll nicht das Verhalten der Ist-Position zur Soll-Position untersucht werden, sondern es werden die verschiedenen Messmethoden der Ist-Position miteinander verglichen:

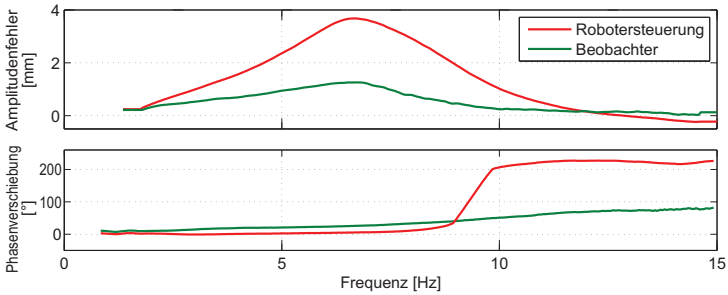
- Die aus der Robotersteuerung ermittelte Position (rot) weicht bei ca. $6,5Hz$ um $3,7mm$ von der tatsächlichen Position (blau) ab. Ab ca. $10Hz$ ergibt sich eine Phasenverschiebung der gemessenen zur tatsächlichen Position von ca. 230° , siehe Abbildung 2.4(a) und (b). Auch bei sprungförmiger Änderung der Roboterposition (siehe Abbildung 2.4(c)), ergibt sich bei dem Positionssignal aus der Robotersteuerung ein in Phase und Betrag stark verfälschtes Signal.
- Die beobachtete Roboterposition (grün) weicht bei $6,5Hz$ nur um $1,2mm$ von der tatsächlichen Position (blau) ab. Die Phasenverschiebung steigt zwischen $0Hz$ und $15Hz$ auf ca. 75° an, siehe Abbildung 2.4(a) und (b). Die beobachtete Sprungantwort (siehe Abbildung 2.4(c), grün), zeigt im Vergleich zur direkt ermittelten Roboterposition eine deutlich höhere Ähnlichkeit mit der tatsächlichen Position (blau gestrichelt).

In Abbildung 2.5 wird die vorige Messung wiederholt, jedoch befindet sich der Roboter nun in einer Konfiguration, die zu deutlich geringerer

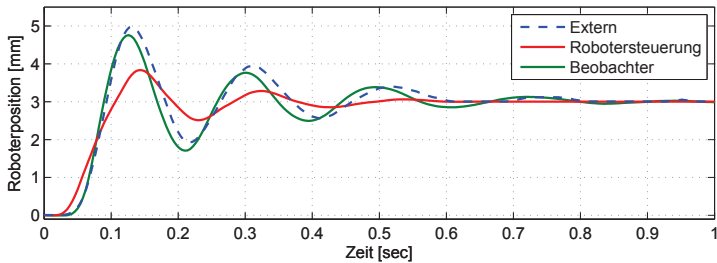
¹⁾Ein Chirpsignal beschreibt ein Signal (z.B. Sinus) mit zeitlich veränderlicher Frequenz $f(t)$.



(a) Vergleich der Messmethoden zur Ermittlung der Ist-Position bei Vorgabe der Soll-Roboterposition als Chirpsignal mit der Frequenz $f(t) = 0,75 \cdot t$ und einer Amplitude von 2mm .



(b) Amplituden- und Phasenfehler des Positionssignales der unterschiedlichen Messmethoden in Abhängigkeit der Signalfrequenz.



(c) Vergleich der Messmethoden bei sprungförmiger Änderung der Roboterposition um 3mm .

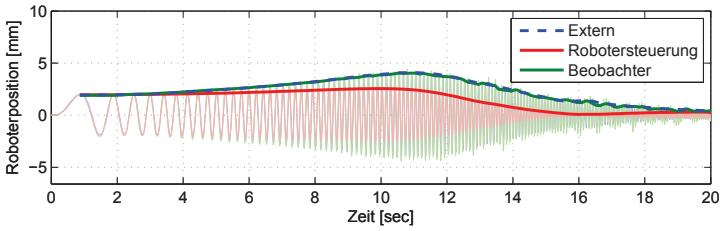
Abbildung 2.4: Vergleich der ermittelten y -Position aus der Robotersteuerung r_y und der beobachteten y -Position $r_{obs,y}$ mit der tatsächlichen y -Position des TCP (mit externem Messsystem ermittelt). Der Frequenzgang (a, b) und die Sprungantwort (c) werden am Roboter mit **hoher Massenträgheit** verglichen (vgl. Abbildung 2.1).

Massenträgheit bezüglich der Gelenke 1-3 bei der gewünschten Bewegung führt. Die Amplitude der beobachteten Position zeigt über den gesamten Frequenzbereich zwischen $0Hz$ und $15Hz$ einen Fehler zur tatsächlichen kartesischen Position von unter $0,2mm$. Die Messung der Position durch die Robotersteuerung zeigt einen Fehler von bis zu $1,7mm$.

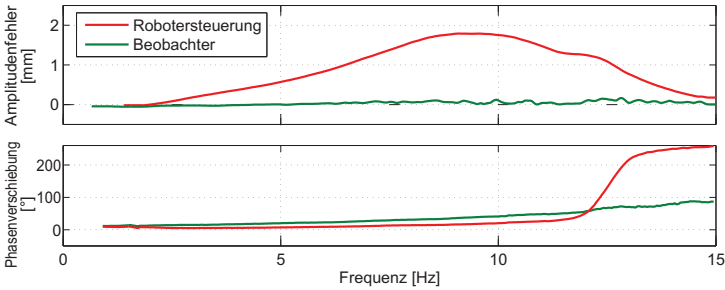
Ein weiterer Vergleich der Messmethoden in einer Konfiguration mit sehr geringer Massenträgheit in Richtung der gewünschten Bewegung ist in Abbildung 2.6 dargestellt. Auch hier zeigt die Amplitude der beobachteten Position über den gesamten Frequenzbereich zwischen $0Hz$ und $15Hz$ einen Fehler zur tatsächlichen Position von unter $0,2mm$, während die Messung der Position durch die Robotersteuerung einen Fehler von bis zu $0,5mm$ erzeugt. Bei der Sprungantwort weist die beobachtete Position bei mittlerer als auch geringer Massenträgheit des Systems einen deutlich geringeren Fehler in Amplitude und Phase zur tatsächlichen Position des TCP auf.

Die Filtereigenschaften des Beobachters bei einer sprungförmigen Anregung des Systems sind in Abbildung 2.7 dargestellt. Die blau gestrichelte Linie zeigt die tatsächliche Bewegung des TCP (gemessen mit dem externen Kamerasystem). Die orange gestrichelte Linie zeigt den Einfluss des tiefpass-gefilterten Positionssignals auf das beobachtete Signal (für diese Messung wurde das Signal des Beschleunigungssensors zu 0 gesetzt). Hochfrequente Anteile sind aus dem Signal herausgefiltert worden. Die orange durchgezogene Linie zeigt den Einfluss des Hochpass-gefilterten Beschleunigungssignals (für diese Messung wurde das Positionssignal des Roboters zu 0 gesetzt). Die stationären Anteile sind nicht mehr im Signal vorhanden. Nur bei gemeinsamer Verwendung des Positions- und des Beschleunigungssignales im Beobachter (grün durchgezogen) ergeben sich die in Abbildung 2.4 bis 2.6 dargestellten Ergebnisse.

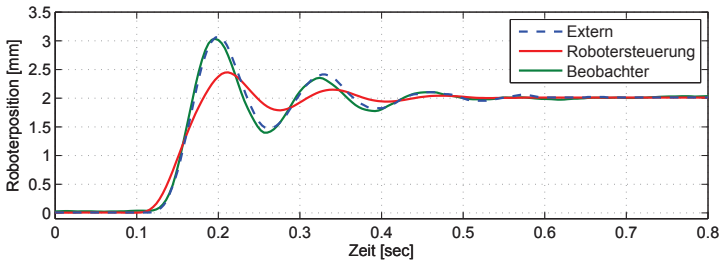
Ein weiterer Vorteil des Beobachters ist in Abbildung 2.8 dargestellt. Sollen Geschwindigkeit bzw. Beschleunigung in einer Regelung mit berücksichtigt werden, so ergibt die erste bzw. zweite Ableitung des Positionssignales aus der Robotersteuerung signifikantes Quantisierungsrauschen, siehe Abbildung 2.8(c) und Abbildung 2.8(e). Dahingegen können im Beobachter die Geschwindigkeit und die Beschleunigung mit deutlich geringerem Signalrauschen direkt ermittelt werden, siehe Abbildung 2.8(d) und Abbildung 2.8(f).



(a) Vergleich der Messmethoden zur Ermittlung der Ist-Position bei Vorgabe der Soll-Roboterposition als Chirpsignal mit der Frequenz $f(t) = 0,75 \cdot t$ und einer Amplitude von 2mm .

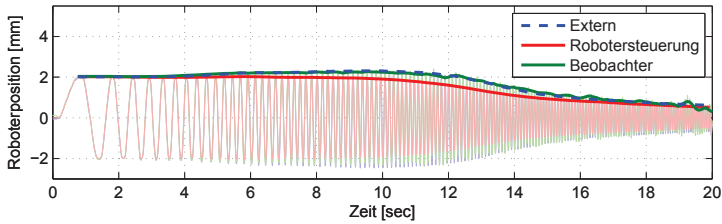


(b) Amplituden- und Phasenfehler des Positionssignales der unterschiedlichen Messmethoden in Abhängigkeit der Signalfrequenz.

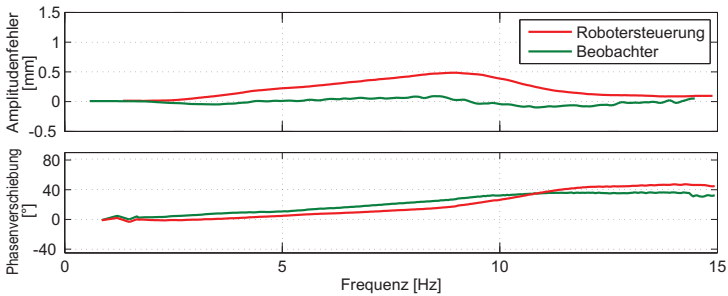


(c) Vergleich der Messmethoden bei sprungförmiger Änderung der Roboterposition um 3mm .

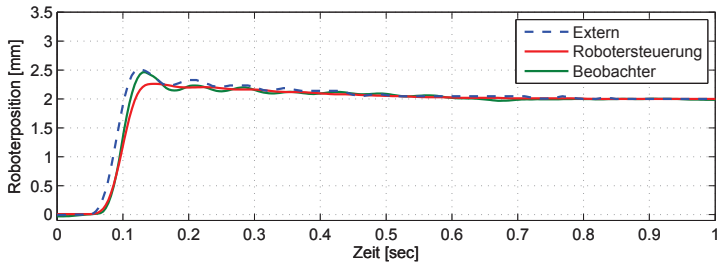
Abbildung 2.5: Vergleich der ermittelten y -Position r_y aus der Robotersteuerung und der beobachteten y -Position $r_{obs,y}$ mit der tatsächlichen y -Position des TCP (mit externem Messsystem ermittelt). Der Frequenzgang (a, b) und die Sprungantwort (c) werden am Roboter mit **mittlerer Massenträgheit** verglichen (vgl. Abbildung 2.1).



(a) Vergleich der Messmethoden zur Ermittlung der Ist-Position bei Vorgabe der Soll-Roboterposition als Chirpsignal mit der Frequenz $f(t) = 0,75 \cdot t$ und einer Amplitude von 2mm .



(b) Amplituden- und Phasenfehler des Positionssignales der unterschiedlichen Messmethoden in Abhängigkeit der Signalfrequenz.



(c) Vergleich der Messmethoden bei sprungförmiger Änderung der Roboterposition um 3mm .

Abbildung 2.6: Vergleich der ermittelten z -Position r_z aus der Robotersteuerung und der beobachteten z -Position $r_{obs,z}$ mit der tatsächlichen z -Position des TCP (mit externem Messsystem ermittelt). Der Frequenzgang (a, b) und die Sprungantwort (c) werden am Roboter mit **geringer Massenträgheit** verglichen (vgl. Abbildung 2.1).

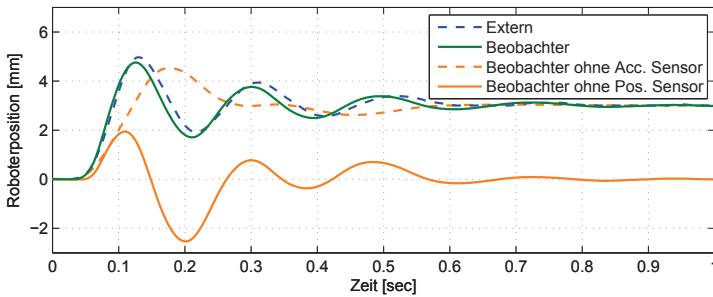
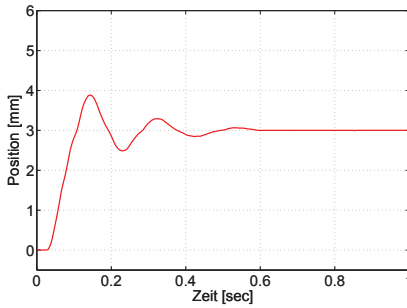


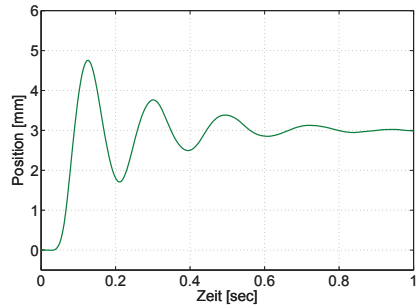
Abbildung 2.7: Ausgang des Beobachters bei Weglassen des Beschleunigungssensors (orange gestrichelt), bzw. bei Weglassen des Positionssignales (orange durchgezogen).

2.1 Zusammenfassung

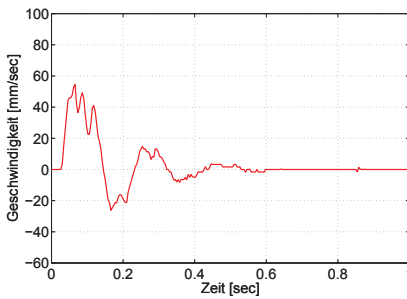
Die Robotersteuerung eines Industrieroboters stellt die kartesische Position des TCP im Interpolationstakt über eine Schnittstelle zur Verfügung. Für positionsbasierte, bildgestützte Verfahren beeinflusst die Exaktheit der TCP Position die Genauigkeit der Bildauswertung mit dem Kamerasystem. Da diese Position aus der Vorwärtstransformation, basierend auf der Messung der Motorwinkel, berechnet wird, ergeben sich Unterschiede zwischen der tatsächlichen Position und der von der Robotersteuerung berechneten Position. In diesem Kapitel wurde der Roboter mit unterschiedlichen Signalformen bewegt, um das Frequenzverhalten und die Sprungantwort des Messsystems zu beurteilen. Die tatsächliche Position des TCP wurde mit einem Kamerasystem erfasst. Es hat sich gezeigt, dass die ermittelte Position von der tatsächlichen Position bei Bewegungen mit hoher Beschleunigung signifikant abweicht. Daher wurde ein Beobachter vorgestellt, der neben der stationären Genauigkeit des Roboterpositionssignales die dynamischen Anteile eines Beschleunigungssensors berücksichtigt. Durch die Beobachterstruktur ist der Fehler zwischen der beobachteten Position und der tatsächlichen Position deutlich geringer. Auch ergibt sich bei Verwendung des Beobachters keine plötzliche Phasenverschiebung.



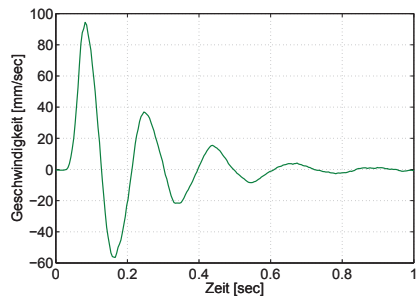
(a) Roterposition



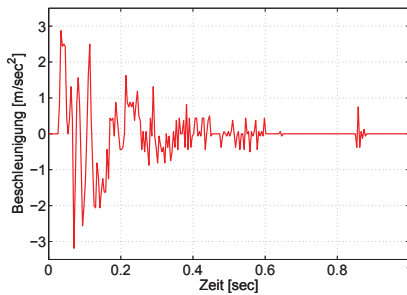
(b) Beobachtete Roterposition



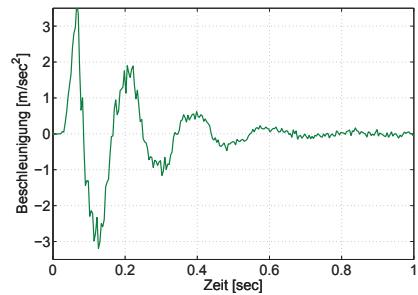
(c) Erste Ableitung der Roterposition



(d) Erste Ableitung der beob. Position



(e) Zweite Ableitung der Roterposition



(f) Zweite Ableitung der beob. Position

Abbildung 2.8: Vergleich zwischen Geschwindigkeit und Beschleunigung durch Ableitung der Position. Links: Signal aus der Robotersteuerung, rechts: Signal aus Beobachter. Der Einfluss von Quantisierungsrauschen durch Ableitung ist beim beobachteten Signal deutlich geringer.

3 Bildgestützte Konturverfolgung

Das Ziel der bildgestützten Konturverfolgung ist die Anpassung von Position und Orientierung des robotergeführten Werkzeuges an eine visuell erfassbare Kontur auf einem Werkstück. Applikationen wie z.B. das Aufbringen von Dichtmasse entlang einer Schweißnaht oder das robotergeführte Nähen oder Entgraten erfordern die Führung des Werkzeuges mit definierter Bahngeschwindigkeit entlang der Kontur. Diese Prozesse erfordern zusätzlich eine definierte Orientierung relativ zur Kontur. In diesem Kapitel wird die rein bildgestützte Konturverfolgung behandelt. Abschnitt 3.1 stellt das Kamerasystem vor, wobei im Gegensatz zum klassischen Visual Servoing nicht direkt am TCP, sondern mit einem Vorlauf gemessen wird. Abschnitt 3.2 befasst sich mit der Konturverfolgung, wobei die Kontur als komplett unbekannt angenommen wird. Neben Position und Orientierung wird die Geschwindigkeit des Werkzeuges automatisch an die Kontur im Vorlauf angepasst, um Winkelgeschwindigkeiten in Kurven und Ecken anzupassen. Abbildung 3.1 zeigt eine Übersicht über die bildgestützte Konturverfolgung mit Kapitelangabe. Teile dieses Kapitels wurden in [KKK⁺11c, KKK⁺12] veröffentlicht.

Die Bewegung des Roboters bei der Konturverfolgung wird im Koordinatensystem $K_{[TCP]}$ des robotergeführten Werkzeuges festgelegt (siehe Abbildung 3.2). Die Bahngeschwindigkeit v_x wird in x -Richtung des Werkzeugkoordinatensystems vorgegeben, wobei die x -Achse von $K_{[TCP]}$ tangential zur Kontur ausgerichtet wird. Die y -Achse wird parallel zur Werkstückebene ausgerichtet. Die z -Achse bildet mit der x - und y -Achse ein Rechtssystem und steht somit senkrecht auf der Kontur. Die Aufgaben definition wird in Tabelle 3.1 zusammengefasst.

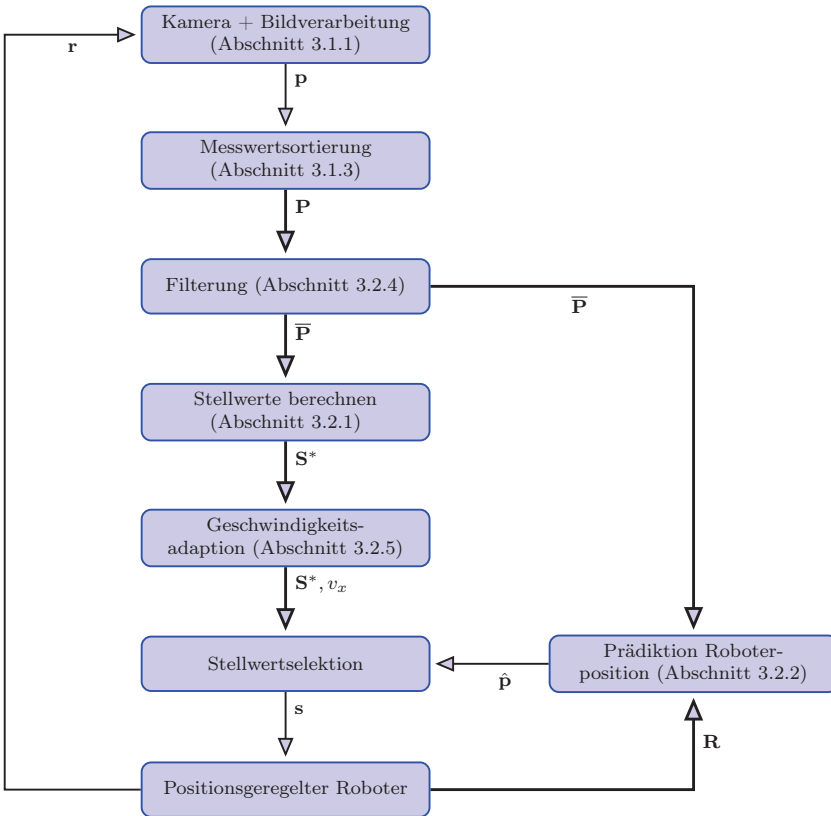


Abbildung 3.1: Übersicht mit Kapitelangabe über die bildgestützte Konturverfolgung. Die gemessenen 3D-Bildmerkmale p jeder Bildaufnahme werden durch geometrische Sortierung in den Merkmalspfad P einsortiert und sowohl zur Generierung des Stellwertpfades S^* als auch zur Prädiktion \hat{p} des Roboterpfades R verwendet. Durch die Prädiktion \hat{p} wird aus dem Stellwertpfad S^* der aktuell benötigte Stellwert s selektiert und an den Roboter gesendet.

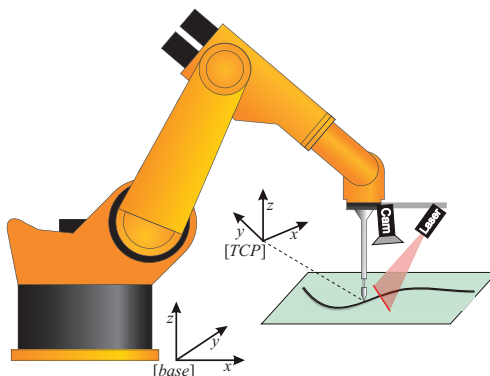


Abbildung 3.2: Systemaufbau zur Konturverfolgung mit vorausschauender Kameraanordnung. Das Kamerasystem (Lasertriangulation) steht in fester Relation zum TCP (eye-in-hand).

Tabelle 3.1: Richtungsdefinition zur bildgestützten Konturverfolgung im Werkzeugkoordinatensystem $K_{[TCP]}$.

| | |
|------------|---|
| x : | Bahngeschwindigkeit v_x entlang der Kontur. |
| y : | Abstand d_y des Werkzeuges zur Kontur (z.B. $d_y = 0mm$). |
| z : | Abstand d_z des Werkzeuges zur Kontur (z.B. $d_z = 0mm$). |
| γ : | Rotation um x -Achse, sodass die y -Achse parallel zur Ebene des Werkstückes liegt. |
| β : | Rotation um y -Achse, sodass die z -Achse senkrecht zur Kontur steht. |
| α : | Rotation um z -Achse, sodass die x -Achse tangential zur Kontur liegt. |

3.1 Bildverarbeitung und Messwertspeicherung

In diesem Abschnitt werden zunächst das Kamerasystem und die 3D-Merkmalsextraktion vorgestellt. Anschließend wird auf die Vorlaufänge der vorausschauenden Messung und auf die geometrische Sortierung von Merkmalen zur Generierung eines Merkmalspfades eingegangen.

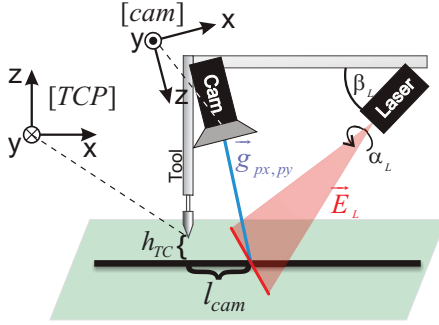


Abbildung 3.3: Aufbau des Laser-Triangulationssensors mit Lichtebene \vec{E}_L und Kamerasehstrahl $\vec{g}_{px,py}$. Die Vorlängfe l_{cam} hängt von der Anordnung, Abstand zur Kontur h_{TC} und der Orientierung des Lasers ab. Die Bewegung erfolgt in x -Richtung des Werkzeugkoordinatensystems $K_{[TCP]}$.

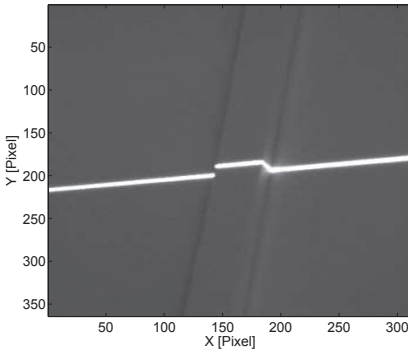
3.1.1 3D-Messprinzip

Zur Erfassung der Kontur wird ein vom Roboter mitgeführter Laser-Triangulationssensor verwendet. Der Aufbau ist schematisch in Abbildung 3.2 und 3.3 dargestellt. Die vom Laser erzeugte Lichtebene projiziert eine Linie auf das Werkstück. Diese Projektion wird von der CCD-Grauwertkamera mit Farbfilter¹⁾ erfasst (siehe Abbildung 3.4(a)). Durch eine Kamerakalibrierung (z.B. [Ott10]) werden die Geradengleichungen aller „Sehstrahlen“ der Kamera $\vec{g}_{px,py}$ in Abhängigkeit der Pixelkoordinaten $\mathbf{p}^{img} = (px, py)^T$ berechnet. Die entsprechende Geradengleichung des Sehstrahles eines Pixels wird jeweils durch zwei dreidimensionale Punkte $\mathbf{p1}_{px,py}$ und $\mathbf{p2}_{px,py}$ in Abhängigkeit der Koordinaten $\mathbf{x} = (x, y, z)^T$ im Kamerakoordinatensystem $K_{[cam]}$ beschrieben:

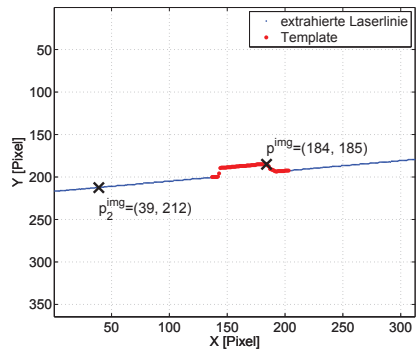
$$\vec{g}_{px,py} : \mathbf{x} = \mathbf{p1}_{px,py} + \lambda \cdot (\mathbf{p2}_{px,py} - \mathbf{p1}_{px,py}) \quad . \quad (3.1)$$

Dabei werden $\mathbf{p1}$ und $\mathbf{p2}$ aus den Daten der Kamerakalibrierung gewonnen. Durch die Kalibrierung wird zudem die Ebenengleichung der Lichtebene \vec{E}_L im Kamerakoordinatensystem $K_{[cam]}$ ermittelt und durch die

¹⁾Der Farbfilter lässt nur Wellenlängen im Bereich des verwendeten Lasers durch, um Störungen durch Umgebungslicht zu reduzieren.



(a) Kamerabild der Kontur im Laserlicht aus der Grauwert CCD-Kamera.



(b) Extrahierte 2D-Laserlinie mit den beiden Merkmalen \mathbf{p}^{img} und \mathbf{p}_2^{img} .

Abbildung 3.4: Auswertung des Kamerabildes. Das Template (rot) wird durch den Benutzer an einem Initialisierungsbild definiert und in nachfolgenden Aufnahmen gesucht. Der Merkmalspfad wird aus \mathbf{p}^{img} zusammengesetzt. \mathbf{p}_2^{img} wird zur Berechnung der Werkstückeebene verwendet (siehe Abschnitt 3.2).

Normale \vec{n}_L und einen Punkt \mathbf{p}_{L0} der Lichtebene folgendermaßen beschreiben²⁾:

$$\vec{E}_L : \vec{n}_L \cdot (\mathbf{x} - \mathbf{p}_{L0}) = 0 \quad . \quad (3.2)$$

In Abbildung 3.4(a) ist das Grauwertbild der Kamera mit Laserprojektion dargestellt. Im Grauwertbild wird spaltenweise das Maximum der Helligkeit gesucht, welches der Laserlinie zuzuordnen ist. Es wird zusätzlich eine Plausibilitätskontrolle durchgeführt, welche die erwartete Dicke der Laserlinie mit einbezieht und das Ergebnis auf Ausreißer untersucht. Das Ergebnis dieser Extraktion ist ein zweidimensionaler Vektor der Laserlinie (Abbildung 3.4(b)) (blau). Der Benutzer definiert zur Initialisierung in einem Beispielbild einen charakteristischen Bereich (rot), welcher in allen darauffolgenden Bildaufnahmen mittels Korrelationsanalyse in der extrahierten Laserlinie gesucht wird. Innerhalb des Templates wird ein Punkt \mathbf{p}^{img} definiert, aus welchem der Merkmalspfad aus aufeinanderfolgenden Bildaufnahmen zusammengesetzt wird.

²⁾Punkte im Raum (Ortsvektoren) werden in dieser Arbeit fett dargestellt (z.B. \mathbf{p}), Richtungsvektoren werden mit Vektorpfeil dargestellt (z.B. \vec{n}_L).

Das Merkmal $\mathbf{p}^{img} = (px, py)^T$ liegt zweidimensional in Bildkoordinaten vor. Dieser Pixelkoordinate ist durch die Kalibrierung eine dreidimensionale Geradengleichung $\vec{g}_{px,py}$ eines Sehstrahles in Kamerakoordinaten zugeordnet (vgl. Gleichung 3.1). Da das Merkmal innerhalb der Laserprojektion gesucht wird, liegt an dieser Stelle ein Schnitt zwischen dem entsprechenden Sehstrahl $\vec{g}_{px,py}$ und der Lichtebeine \vec{E}_L vor. Der Schnittpunkt von Gerade und Ebene ergibt ein dreidimensionales Merkmal \mathbf{p}^{cam} in Kamerakoordinaten:

$$\mathbf{p}^{cam} = \mathbf{p1}_{px,py} + \lambda \cdot (\mathbf{p2}_{px,py} - \mathbf{p1}_{px,py}),$$

$$\text{mit } \lambda = \left(\frac{\vec{n}_L \cdot (\mathbf{p}_{L0} - \mathbf{p1}_{px,py})}{\vec{n}_L \cdot (\mathbf{p2}_{px,py} - \mathbf{p1}_{px,py})} \right). \quad (3.3)$$

Zur bildgestützten Konturverfolgung werden Merkmale in einem feststehenden Koordinatensystem $K_{[base]}$ benötigt. Aus der Robotersteuerung ist zu jedem Zeitpunkt der Zusammenhang zwischen dem Werkzeugkoordinatensystem $K_{[TCP]}$ und dem Basiskoordinatensystem $K_{[base]}$ bekannt. Bei der Kamerakalibrierung wird der Zusammenhang zwischen dem Kamerakoordinatensystem $K_{[cam]}$ und dem Werkzeugkoordinatensystem ermittelt. Dadurch kann ein Punkt \mathbf{p}^{cam} aus dem Kamerakoordinatensystem folgendermaßen in das Basiskoordinatensystem transformiert werden:

$$\mathbf{p}^{base} = \mathbf{p} = \mathbf{T}_{base}^{TCP} \cdot \mathbf{T}_{TCP}^{cam} \cdot \mathbf{p}^{cam}. \quad (3.4)$$

Ein Punkt $\mathbf{p} = (p_x, p_y, p_z)^T$ beschreibt nachfolgend immer einen dreidimensionalen Punkt im Basiskoordinatensystem $K_{[base]}$, sofern nicht ausdrücklich anders gekennzeichnet (z.B. \mathbf{p}^{TCP}).

Bildaufnahme und Bildverarbeitung benötigen die Zeit τ_{IP} zur Berechnung von \mathbf{p} . Dazu zählen die Integrationszeit des CCD-Sensors, Übertragungsverzögerung und Rechenzeit. Das zum Zeitpunkt t berechnete Merkmal in Kamerakoordinaten muss daher zur Umrechnung in dreidimensionale Messwerte mit der homogenen Transformation \mathbf{T}_{base}^{TCP} zum Zeitpunkt $t - \tau_{IP}$ verrechnet werden. Die Bildverarbeitung in den Experimenten erzeugt abhängig von Belichtungsdauer und Merkmal eine Verzögerung von z.B. $\tau_{IP} = 24ms$. Die Berücksichtigung der Verzögerung ist in Abbildung 3.5 dargestellt. Bei dieser Messung wird eine geradlinige Kontur unter folgenden Randbedingungen gemessen:

1. Der Roboter führt das Kamerasystem in einer linearen Bewegung über die geradlinige Kontur (blau).

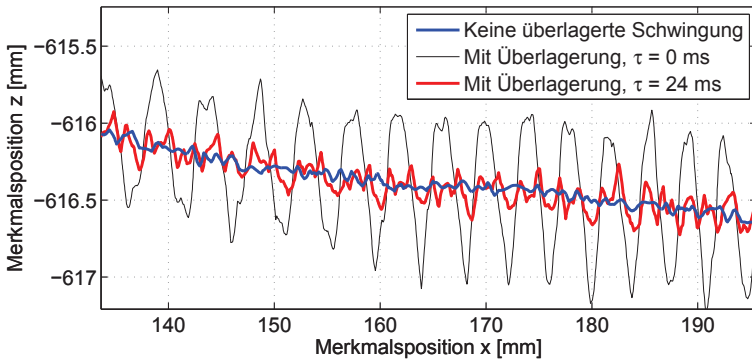


Abbildung 3.5: Einfluss der Zeitverzögerung τ_{IP} auf die Bildverarbeitung. Die Berücksichtigung der Zeitverzögerung reduziert den Fehler der Berechnung der dreidimensionalen Merkmalsposition. Als Störgröße wurde eine sinusförmige Bewegung des Roboters mit ca. $4Hz$ vorgegeben.

2. Quer zur linearen Bewegung wird eine sinusförmige Bewegung mit einer Amplitude von $\pm 2mm$ bei einer Frequenz zwischen $0Hz$ und $5Hz$ überlagert. Dabei wird die Roboterposition
 - a) direkt zur Transformation der Bilddaten verwendet (schwarz).
 - b) um $\tau_{IP} = 24ms$ verzögert, bevor sie zur Transformation der Bilddaten verwendet wird (rot).

Ohne Überlagerung der Roboterbewegung quer zur Kontur wird das Merkmal wie erwartet aufgezeichnet (blau). Bei überlagelter Bewegung ohne Berücksichtigung der Zeitverzögerung τ_{IP} ergibt sich ein Fehler von ca. $\pm 0.5mm$ im dreidimensionalen Merkmalspfad (schwarz). Diese Störung wird stark reduziert, wenn die **verzögerte** Roboterposition zur Merkmalsberechnung verwendet wird (rot), da dies die Bildaufnahme mit der Kameraposition zum Zeitpunkt der Bildaufnahme synchronisiert.

3.1.2 Messwertaufzeichnung bei veränderlicher Vorlauflänge

Das Merkmal \mathbf{p} wird aufgrund der Anordnung des Kamerasystems mit einem Vorlauf von $l_{cam} \approx 50mm$ vor dem TCP im Schnittpunkt zwischen

Laserprojektion und Kontur gemessen (siehe Abbildung 3.3). Die Vorlauf-
länge hängt von folgenden Einflüssen ab:

- Aufbau und Anbringung des Kamerasystems am Roboter.
- Abstand des Werkzeugkoordinatensystems zur Kontur.
- Orientierung des Werkzeugkoordinatensystems relativ zur Kontur.

Abstand und Orientierung zwischen Kamerasystem und Werkstück sind während des Prozesses entsprechend der Form der Kontur veränderlich. Folglich muss die Vorlauf- l_{cam} als veränderlich vorausgesetzt werden. Der Einfluss der veränderlichen Vorlauf- l_{cam} auf die Messwertverarbeitung wird in diesem Abschnitt untersucht.

Entlang einer Kontur werden einzelne 3D-Merkmalpunkte \mathbf{p} ermittelt. Eigenschaften des Pfades, wie z.B. die Orientierung und Segmentlängen, werden aus einer Sequenz \mathbf{P}

$$\mathbf{P} = \{\mathbf{p}_0, \mathbf{p}_1, \dots, \mathbf{p}_N\}, \quad \mathbf{p}_j = (p_{xj}, p_{yj}, p_{zj})^T, \quad j = 0, \dots, N \quad (3.5)$$

von Messwerten \mathbf{p}_j berechnet. Die Länge l eines Segmentes zwischen zwei Punkten \mathbf{p}_m und \mathbf{p}_n wird folgendermaßen berechnet:

$$l = \sum_{j=m}^{n-1} \|\mathbf{p}_{j+1} - \mathbf{p}_j\| \quad . \quad (3.6)$$

Liegen einzelne Messwerte \mathbf{p}_j in falscher räumlicher Reihenfolge entlang der Kontur vor („rückläufige Abtastung“), führt Gleichung 3.6 zu einer fehlerhaften Berechnung der Länge l , sodass der Abstand zwischen zwei Punkten \mathbf{p}_m und \mathbf{p}_n entlang der Kontur falsch berechnet werden würde.

Zu der rückläufigen Abtastung kommt es, wenn sich die Vorlauf- l_{cam} während einer Bewegung so stark verändert, dass sich der Schnittpunkt zwischen Laser und Kontur trotz Vorwärtsbewegung des TCP entgegen dieser Richtung bewegt. Häufig kommt es in Experimenten im Bereich von starken Krümmungen oder Ecken zu der rückläufigen Abtastung. Diese Problematik ist in Abbildung 3.6 dargestellt. Die schwarze Linie stellt eine Kontur mit starker Krümmung in der x - y -Ebene dar. Es werden sechs Positionen T_i ($i = 1, \dots, 6$) des TCP mit zugehöriger Orientierung dargestellt. Die rote Laserlinie steht in fester Relation zum TCP, welcher mit der x -Achse tangential an der Kontur ausgerichtet wird. Das Kamerasystem erfasst das Merkmal im Schnittpunkt L_i zwischen Laserlinie und Kontur. Während der Annäherung an die Kurve ($i = 1, \dots, 3$) entfernt sich der

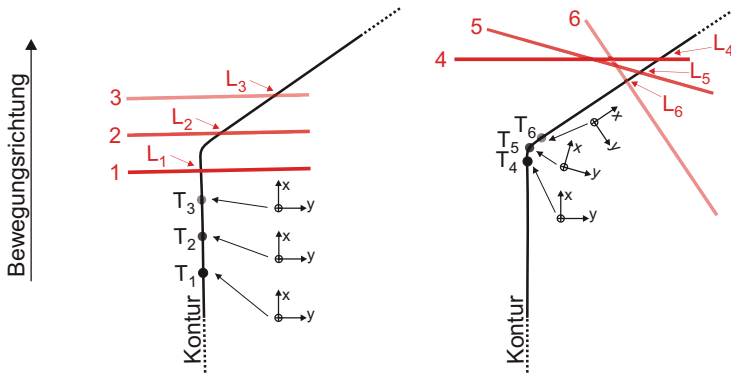


Abbildung 3.6: Der Abstand zwischen TCP T_i und Messpunkt der Kamera (Schnittpunkt L_i zwischen Laser und Kontur) ändert sich in Abhängigkeit der TCP Position. Laser und TCP stehen zueinander in fester Relation und sind für sechs Positionen $i = 1, \dots, 6$ dargestellt.

Schnittpunkt L_i aufgrund der Form der Kontur vom TCP. Bei der Umorientierung in der Kurve ($i = 4, \dots, 6$) verschiebt sich der Schnittpunkt L_i wieder näher an den TCP. Durch die Geometrie verkürzt sich der Abstand zwischen T_i und L_i so stark, dass trotz der Vorwärtsbewegung des TCP entlang der Kontur für einen kurzen Zeitraum während der Umorientierung die rückläufige Abtastung eintritt.

Die Erfassung einer Kontur an einer Ecke wird in Abbildung 3.7 gezeigt. Ein Teil des Pfades liegt in rückläufiger Reihenfolge vor und wird daraufhin mehrfach aufgezeichnet (die Tatsache, dass die Messwerte in der Überlappung nicht exakt auf derselben Linie verlaufen, liegt an der ungenauen Kalibrierung des Kamerasystems). Um Orientierungen entlang des Pfades korrekt zu berechnen, muss der mehrfach aufgezeichnete Bereich entsprechend geometrisch sortiert werden.

3.1.3 Geometrische Sortierung der Messwerte

Um den Einfluss durch Messfehler und durch rückläufige Erfassung von Merkmalen zu reduzieren, werden neu erfasste Merkmale \mathbf{p}_{new} nicht sequentiell an den Merkmalspfad \mathbf{P} angehängt, sondern durch einen Sortieralgorithmus validiert und geometrisch einsortiert. Die Sortierung be-

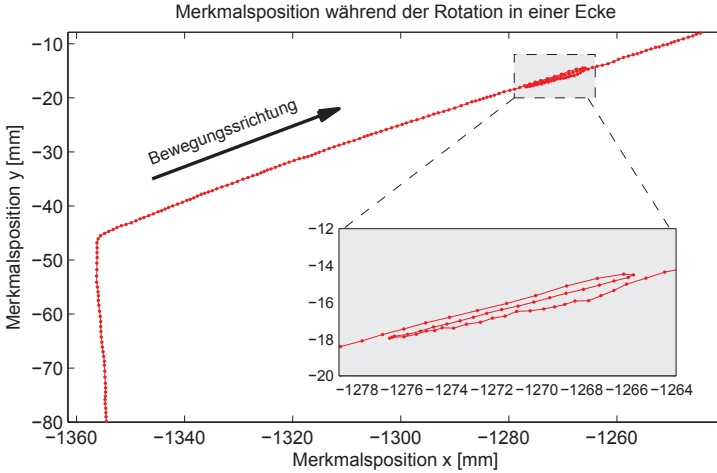


Abbildung 3.7: Ein Pfadsegment wird während der Rotation um die z -Achse des Werkzeuges mehrfach aufgezeichnet. Die Stützstellen liegen nun nicht in räumlich korrekter Reihenfolge entlang der Kontur vor.

ruht auf der Annahme eines stetigen Verlaufes in den Messwerten (Unstetigkeiten, wie z.B. eine Ecke sind jedoch im Pfad \mathbf{P} erwünscht und werden gesondert berücksichtigt). Die Richtung der Kontur wird aus Messwerten bestimmt; Messrauschen, speziell bei sehr langsamer Bewegung oder Stillstand, führt zu Fehlern, da Richtungen aus nahezu identischen Merkmalen berechnet werden. Um dies zu vermeiden, werden neue Messwerte \mathbf{p}_{new} nur dann in \mathbf{P} einsortiert, wenn der Abstand d_{next} zu den benachbarten Messwerten einen Minimalabstand d_{min} überschreitet:

$$d_{next} \stackrel{!}{\geq} d_{min} \quad \text{z.B. } d_{min} = 0,1mm \quad . \quad (3.7)$$

Durch den Minimalabstand wird der Rechenaufwand durch Datenreduktion verringert. Genügt ein neu aufgezeichneter Punkt \mathbf{p}_{new} dem Abstandskriterium nach Gleichung 3.7, wird daraufhin versucht, \mathbf{p}_{new} in den Merkmalspfad \mathbf{P} zu integrieren. Abbildung 3.8 veranschaulicht den Sortieralgorithmus. Der neu aufgezeichnete Punkt ist im Bereich des Kameravorlaufs aufgezeichnet worden. Daher wird der mögliche Bereich zur Einsortierung neu aufgezeichneter Merkmale in den vorhandenen Pfad \mathbf{P} durch die maximale Vorlauflänge $l_{cam,max}$ beschränkt (die Vorlauflänge ist in

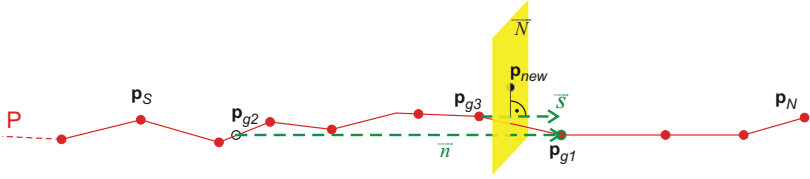


Abbildung 3.8: Einsortierung eines neuen Messwertes \mathbf{p}_{new} in den Merkmalspfad \mathbf{P} . Der Index zur Einsortierung wird aus dem Schnitt zwischen \mathbf{P} und der Normalenebene \vec{N} des Pfades durch den neuen Punkt \mathbf{p}_{new} gewonnen.

Abbildung 3.3 dargestellt). Ausgehend vom aktuellen Endpunkt \mathbf{p}_N des Merkmalspfades wird der Startpunkt \mathbf{p}_S des Algorithmus gesucht, mit

$$\|\mathbf{p}_N - \mathbf{p}_S\| = l_{cam,max} \quad . \quad (3.8)$$

Bei \mathbf{p}_S beginnend wird versucht, den neu aufgezeichneten Punkt \mathbf{p}_{new} in den Merkmalspfad einzusortieren. Dazu wird ein Punkt \mathbf{p}_{g1} schrittweise über alle nachfolgenden Elemente im Merkmalspfad von \mathbf{p}_S bis \mathbf{p}_N verschoben, wobei jeweils ein Richtungsvektor $\vec{n} = \mathbf{p}_{g1} - \mathbf{p}_{g2}$ definiert wird (\mathbf{p}_{g2} wird im Abstand von ca. 5mm vor \mathbf{p}_{g1} ermittelt). \vec{n} spiegelt somit die Richtung des aktuell betrachteten Pfadsegmentes wider. Eine Ebene \vec{N} wird mit \vec{n} als Normalenvektor und \mathbf{p}_{new} als Stützpunkt aufgespannt. Zudem wird ein Geradensegment \vec{s} in Richtung von \vec{n} mit Stützpunkt \mathbf{p}_{g3} definiert (\mathbf{p}_{g3} ist der nächste Nachbarnpunkt von \mathbf{p}_{g1}), wobei der Richtungsvektor \vec{s} auf die Länge $\|\mathbf{p}_{g3} - \mathbf{p}_{g1}\|$ normiert wird:

$$\vec{s}: \quad \mathbf{x} = \mathbf{p}_{g3} + \lambda \cdot \frac{\vec{n}}{\|\vec{n}\|} \cdot \|\mathbf{p}_{g1} - \mathbf{p}_{g3}\| = \mathbf{p}_{g3} + \lambda \cdot \vec{n}_n \quad (3.9)$$

$$\vec{N}: \quad \vec{n}_n \cdot (\mathbf{x} - \mathbf{p}_{new}) = 0 \quad . \quad (3.10)$$

Einsetzen der Geradengleichung 3.9 in die Ebenengleichung 3.10 ergibt den Schnittpunkt mit geringstem Normalenabstand zwischen \mathbf{p}_{new} und \vec{s} :

$$\lambda = \frac{\vec{n}_n \cdot (\mathbf{p}_{new} - \mathbf{p}_{g3})}{\vec{n}_n \cdot \vec{n}_n} \quad . \quad (3.11)$$

Abhängig von λ wird eine Fallunterscheidung durchgeführt:

- **Fall 1:** $\lambda < 0$: Der neue Punkt \mathbf{p}_{new} kann nicht einsortiert werden und wird verworfen (z.B. durch Messfehler).

- **Fall 2:** $0 \leq \lambda \leq 1$: Der neue Punkt \mathbf{p}_{new} liegt zwischen \mathbf{p}_{g3} und \mathbf{p}_{g1} . Dieser Punkt wird im Speicher \mathbf{P} zwischen \mathbf{p}_{g1} und \mathbf{p}_{g3} einsortiert, sofern es sich nicht um einen Ausreißer handelt. Ausreißer werden über die resultierenden Knickwinkel im Pfad überprüft. Mit z.B. $\alpha_{Kmax} = 30^\circ$ wird \mathbf{p}_{new} einsortiert falls: (Der Winkel $\alpha_{Kmax} = 30^\circ$ wurde durch Versuche ermittelt. Je nach erwarteter Krümmung und Signalrauschen ist dieser Wert einem Experiment anzupassen)

$$\begin{aligned} \angle(\mathbf{p}_{g3}, \mathbf{p}_{new}, \mathbf{p}_{g1}) &< \alpha_{Kmax} \\ \text{und } \angle(\mathbf{p}_{g3-1}, \mathbf{p}_{g3}, \mathbf{p}_{new}) &< \alpha_{Kmax} \\ \text{und } \angle(\mathbf{p}_{new}, \mathbf{p}_{g1}, \mathbf{p}_{g1+1}) &< \alpha_{Kmax} \end{aligned} \quad (3.12)$$

- **Fall 3a:** $\lambda > 1$ und $\mathbf{p}_{g1} \neq \mathbf{p}_N$: Der Punkt \mathbf{p}_{new} kann nicht zwischen \mathbf{p}_{g1} und \mathbf{p}_{g2} sortiert werden. Der Suchalgorithmus fährt mit der nächsten Iteration von \mathbf{p}_{g1} entlang des Pfades in Gleichung 3.9 und Gleichung 3.10 fort.
- **Fall 3b:** $\lambda > 1$ und $\mathbf{p}_{g1} = \mathbf{p}_N$: Der Punkt \mathbf{p}_{new} liegt in der positiven Verlängerung der beiden letzten Punkte im Pfad (also am Ende des Pfades). Analog zu Fall 2 wird der resultierende Knickwinkel im Pfad überprüft und \mathbf{p}_{new} nur an das Pfadende angehängt, wenn gilt:

$$\angle(\mathbf{p}_{g2}, \mathbf{p}_{g1}, \mathbf{p}_{new}) < \alpha_{Kmax} \quad . \quad (3.13)$$

Einsortierung von Ecken

Das direkte Verwerfen von Messpunkten nach Gleichung 3.13 führt dazu, dass an tatsächlichen Ecken entlang der Kontur keine Messwerte in den Merkmalspfad sortiert werden. Daher wird zusätzlich überprüft, ob die Merkmale in einer tatsächlichen Ecke liegen, deren Winkel größer ist als α_{Kmax} . Abbildung 3.9 verdeutlicht diesen Fall. Die aussortierten Punkte (blau) werden in einen Zwischenspeicher gelegt („on hold“). Wenn z.B. 10 aufeinanderfolgende Punkte mit wachsender Distanz zu \mathbf{p}_N und ungefähr dem gleichen Winkel α_K in den Zwischenspeicher gelegt werden, kann davon ausgegangen werden, dass es sich hierbei um keinen Messfehler handelt. Die zwischengespeicherten Elemente werden daraufhin an den Pfad \mathbf{P} angehängt. Nachfolgende Messwerte \mathbf{p} liegen daraufhin wieder auf einem stetigen Pfad.

Abbildung 3.10 zeigt die Sortierung an einer Ecke und die Sortierung bei Aufzeichnung in falscher räumlicher Reihenfolge entlang der Kontur.

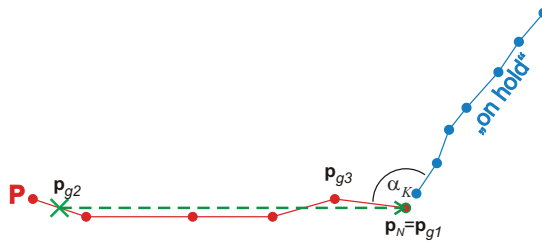


Abbildung 3.9: Einsortierung von Messwerten an einer Ecke.

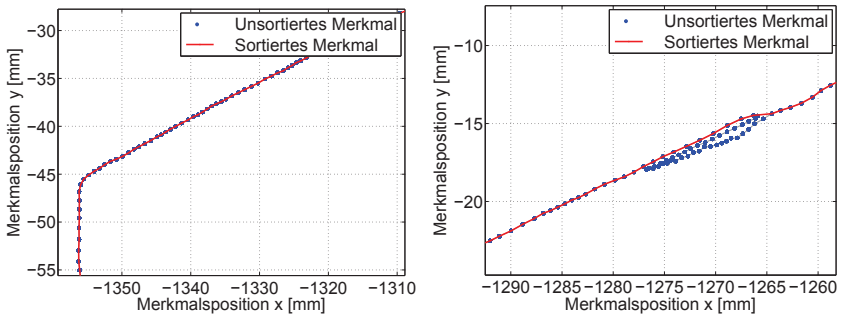


Abbildung 3.10: Einsortierung von Merkmalen an einer Ecke und bei falscher räumlicher Aufzeichnung entlang der Kontur.

An der Ecke werden alle Merkmalspunkte übernommen; bei der mehrfachen Messung eines Segmentes durch Änderung der Kameraorientierung werden die Punkte so übernommen, dass ein Pfad in korrekter Sortierung vorliegt. Der sortierte Merkmalspfad \mathbf{P} bildet die Basis für die nachfolgend beschriebene prädiktive Konturverfolgung.

3.2 Algorithmus zur prädiktiven Konturverfolgung

Ziel der Konturverfolgung ist die Anpassung der Orientierung und der Position des robotergeführten Werkzeuges entlang der sortierten Messwerte \mathbf{P} der zu verfolgenden Kontur auf einem Werkstück. D.h., der Roboter soll

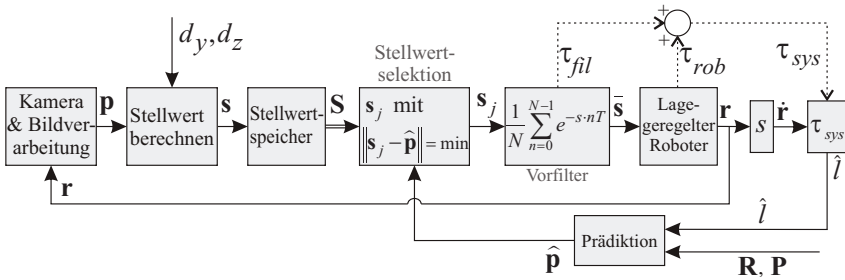


Abbildung 3.11: Regelungsstruktur der bildbasierten prädiktiven Regelung zur Konturverfolgung.

entlang der Kontur geführt werden, sodass neben der Anpassung der Orientierung bei vorgegebener Geschwindigkeit v_x der Abstand $d_y = 0mm$ und $d_z = 0mm$ zwischen \mathbf{P} und dem Pfad \mathbf{R} des Roboters eingehalten wird (vgl. Aufgabendefinition in Tabelle 3.1, Seite 57). Der Pfad

$$\mathbf{R} = \{\mathbf{r}_0, \mathbf{r}_1, \dots, \mathbf{r}_M\}, \quad (3.14)$$

$$\mathbf{r}_k = (r_{xk}, r_{yk}, r_{zk}, r_{\gamma k}, r_{\beta k}, r_{\alpha k})^T, \quad k = 0, 1, \dots, M$$

des Roboters setzt sich aus der kartesischen Position x, y, z und den Rotationen γ, β, α um die x -, y - bzw. z -Achse des Basiskoordinatensystems zusammen, welche zu den diskreten Abtastzeitpunkten k im Interpolationstakt $T_{IPO} = 4ms$ aus der Robotersteuerung ermittelt werden.

In jedem Interpolationstakt T_{IPO} müssen kartesische Stellwerte

$$\mathbf{s}_k = (s_{xk}, s_{yk}, s_{zk}, s_{\gamma k}, s_{\beta k}, s_{\alpha k})^T, \quad k = 0, 1, \dots, M \quad (3.15)$$

erzeugt werden, sodass der Roboterpfad \mathbf{R} an die Merkmale \mathbf{P} angepasst wird. Das Blockschaltbild zur bildgestützten Konturverfolgung ist in Abbildung 3.11 dargestellt. Die einzelnen Komponenten werden in den nachfolgenden Abschnitten erläutert.

3.2.1 Berechnung rotatorischer Stellwerte im Vorlauf

Die kartesischen Stellwerte für den Roboter werden aus den Messwerten der vorausschauenden Kamera durch die geometrische Relation zwischen

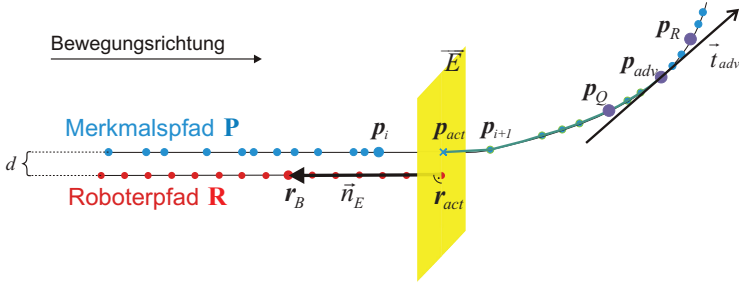


Abbildung 3.12: Herstellen der geometrischen Relation zwischen Roboterpfad und Merkmalspfad mittels Korrespondenzpunkt \mathbf{p}_{act} zum aktuellen Roboterpunkt \mathbf{r}_{act} . Zwischen \mathbf{p}_Q und \mathbf{p}_R wird eine Tangente \vec{t}_{adv} zur Bestimmung der Orientierung im Vorlauf approximiert.

dem Roboterpfad **R** und dem Merkmalspfad **P** berechnet. **R** liegt in zeitlich diskreter Abtastung vor; **P** liegt in räumlicher Sortierung vor. Um einen Bezug zwischen den beiden Pfaden herzustellen, muss die geometrische Relation berechnet werden. Dazu wird in jedem Interpolationsschritt zur aktuellen Roboterposition $\mathbf{r}_{act} = \mathbf{r}_M$ der Merkmalspunkt \mathbf{p}_{act} mit minimalem Normalenabstand gesucht: Es wird von \mathbf{r}_{act} ein Vektor \vec{n}_E zu einem $5mm$ entfernten liegenden Punkt \mathbf{r}_B gebildet, um eine Normalenebene \bar{E} auf dem Roboterpfad zu beschreiben (siehe Abbildung 3.12):

$$\bar{E}: \quad \vec{n}_E \cdot (\mathbf{x} - \mathbf{r}_{act}) = 0 \quad \text{mit } \vec{n}_E = \mathbf{r}_B - \mathbf{r}_{act} \quad . \quad (3.16)$$

In einer Schleife wird die Verbindung \vec{g}_i zwischen zwei benachbarten Merkmalen \mathbf{p}_i und \mathbf{p}_{i+1} gebildet:

$$\vec{g}_i: \quad \mathbf{x} = \mathbf{p}_i + \lambda \cdot \vec{n}_{g,i} \quad \text{mit } \vec{n}_{g,i} = \mathbf{p}_{i+1} - \mathbf{p}_i \quad . \quad (3.17)$$

Einsetzen der Gleichung 3.17 in Gleichung 3.16 ergibt den Schnittpunkt \mathbf{p}_{act} (siehe Abbildung 3.12) zwischen **P** und der Normalenebene auf **R**, falls der Schnittpunkt zwischen \mathbf{p}_i und \mathbf{p}_{i+1} liegt:

$$\begin{aligned} \lambda_i &= \frac{\vec{n}_E \cdot (\mathbf{r}_{act} - \mathbf{p}_i)}{\vec{n}_E \cdot \vec{n}_{g,i}} \\ \Rightarrow \mathbf{p}_{act} &= \mathbf{p}_i + \lambda_i \cdot \vec{n}_{g,i} \quad \text{falls } 0 \leq \lambda_i \leq 1 \quad . \end{aligned} \quad (3.18)$$

Zur vorausschauenden Generierung der Stellwerte muss der zukünftige Verlauf der Roboterbewegung bekannt sein. Es ist aber ausschließlich der zukünftige Verlauf der Merkmale bekannt. Durch die gefundene geometrische Relation zwischen \mathbf{P} und \mathbf{R} kann jedoch der zukünftige Verlauf der Bewegung anhand der gemessenen Merkmale prädiziert werden, da das Ziel der Regelung darin besteht, den Roboterpfad an den Merkmalspfad anzupassen. Daher wird in jedem Interpolationsschritt auf \mathbf{P} , beginnend bei \mathbf{p}_{act} , eine konstante Strecke der Länge $l_{adv} < l_{cam}$ vorgelaufen, um im Vorlauf den Punkt \mathbf{p}_{adv} zu definieren. Die Berechnung der zukünftigen Stellwerte wird im Vorlaufpunkt \mathbf{p}_{adv} durchgeführt. Um \mathbf{p}_{adv} eine Ausrichtung zuzuordnen, wird in diesem Punkt eine Tangente \vec{t}_{adv} an die Kontur approximiert. Eine Möglichkeit ist die Approximation durch eine Sekante der Länge l_{sec} zwischen den Nachbarpunkten \mathbf{p}_Q und \mathbf{p}_R (siehe Abbildung 3.12). Diese Sekante definiert die Richtung \vec{t}_{adv} der Tangente im Vorlaufpunkt \mathbf{p}_{adv} folgendermaßen:

$$\vec{t}_{adv} = \mathbf{p}_R - \mathbf{p}_Q \quad . \quad (3.19)$$

Laut Tabelle 3.1 (Seite 57) wird die x -Achse des Werkzeugkoordinatensystems $K_{[TCP]}$ parallel zu \vec{t}_{adv} ausgerichtet. Aus \vec{t}_{adv} lässt sich jedoch nicht die Rotation γ um die x -Achse bestimmen. Für die Bestimmung dieser Rotation wird der bereits in Abbildung 3.4(b) (Seite 59) dargestellte zweite Merkmalspunkt \mathbf{p}_2^{img} verwendet. Dieser Punkt liegt in einer konstanten Entfernung zu \mathbf{p}^{img} innerhalb der Laserlinie und wird mithilfe der Kalibrierdaten in den dreidimensionalen Punkt

$$\mathbf{p}'_j = (p'_{xj}, p'_{yj}, p'_{zj})^T, \quad j = 0, 1, \dots, N \quad (3.20)$$

umgerechnet. Dadurch ist zu jedem Bildmerkmal \mathbf{p} ein zweites Bildmerkmal \mathbf{p}' verfügbar. Nun kann die Orientierung der Werkstückeebene um die Kontur im Vorlauf beschrieben werden, indem der Normalenvektor

$$\vec{z}_{plane} = \frac{\vec{t}_{adv} \times (\mathbf{p}'_{adv} - \mathbf{p}_{adv})}{|\vec{t}_{adv} \times (\mathbf{p}'_{adv} - \mathbf{p}_{adv})|} \quad (3.21)$$

der Ebene aus der Tangente und dem Merkmalspaar im Vorlauf berechnet wird. Dies ermöglicht in jedem Zyklus k die Berechnung der homogenen

Transformationsmatrix $\mathbf{T}_{base}^{S,k}$ zur Beschreibung der gewünschten Orientierung von $K_{[TCP]}$ im Vorlaufpunkt $\mathbf{p}_{adv,k}$

$$\mathbf{T}_{base}^{S,k} = \begin{pmatrix} n_{kx} & o_{kx} & a_{kx} & 0 \\ n_{ky} & o_{ky} & a_{ky} & 0 \\ n_{kz} & o_{kz} & a_{kz} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad \text{mit} \quad \begin{aligned} \vec{n}_k &= \vec{t}_{adv,k} / |\vec{t}_{adv,k}| \\ \vec{o}_k &= \vec{z}_{plane,k} \times \vec{n}_k \\ \vec{a}_k &= \vec{n}_k \times \vec{o}_k \end{aligned} \quad . \quad (3.22)$$

Die Robotersteuerung des KUKA KR-60 erwartet zur Vorgabe rotatorischer Stellwerte keine Transformationsmatrix in Basiskoordinaten, sondern Eulerwinkel mit „ZYX“-Konvention in Bezug auf die Stellung des Werkzeugkoordinatensystems zum diskreten Zeitpunkt $k = 0$ (Startframe $K_{[start]}$). Die Transformationsmatrix

$$\mathbf{T}_{base}^{start} = Rot_z(\alpha_0) \cdot Rot_y(\beta_0) \cdot Rot_x(\gamma_0) \quad (3.23)$$

$$\mathbf{T}_{start}^{base} = (\mathbf{T}_{base}^{start})^{-1} \quad (3.24)$$

vom Basissystem zum Startframe wird aus den Winkeln α_0 , β_0 und γ_0 zum Zeitpunkt $k = 0$ berechnet. Damit kann die Transformationsmatrix der gewünschten Werkzeugorientierung in jedem diskreten Zeitschritt k im Koordinatensystem des Startframes $K_{[start]}$ wiedergegeben werden:

$$\mathbf{T}_{start}^{S,k} = \mathbf{T}_{start}^{base} \cdot \mathbf{T}_{base}^{S,k} \quad . \quad (3.25)$$

Um die kartesischen Sollwerte an die Robotersteuerung zu senden, müssen die Eulerwinkel in ZYX-Konvention durch eine Inverse Euler Transformation ermittelt werden [SSVO09]:

$$(s_{\alpha k}, s_{\beta k}, s_{\gamma k})^T = f \left(\mathbf{T}_{start}^{S,k} \right) \quad . \quad (3.26)$$

Gleichung 3.26 beschreibt nun die gewünschte Orientierung des Werkzeuges im Vorlaufpunkt $\mathbf{p}_{adv,k}$ zum Zeitpunkt k und wird im Stellwertpfad

$$\begin{aligned} \mathbf{S}^* &= \{\mathbf{s}_0^*, \mathbf{s}_1^*, \dots, \mathbf{s}_N^*\}, \\ \mathbf{s}_k^* &= (\mathbf{p}_{adv,k}, s_{\gamma k}, s_{\beta k}, s_{\alpha k})^T, \quad k = 0, 1, \dots, N \quad . \end{aligned} \quad (3.27)$$

gespeichert. Die in \mathbf{S}^* gespeicherten Stellwerte wurden im Vorlauf ermittelt. D.h., es muss nun zum richtigen Zeitpunkt ein Element aus \mathbf{S}^* selektiert werden, um die gewünschte Orientierung des Roboters vorzugeben. Dies wird mithilfe der nachfolgend beschriebenen Prädiktion realisiert.

3.2.2 Prädiktion zur Selektion des Stellwertes

Die Bewegung des Roboters wird durch die Vorgabe der Geschwindigkeit v_x in x -Richtung des Werkzeugkoordinatensystems definiert. Aus den Orientierungen des Werkzeugkoordinatensystems $K_{[TCP]}$ während der Bewegung ergibt sich die aktuelle kartesische Position in Basiskoordinaten zum diskreten Zeitpunkt N folgendermaßen:

$$\begin{pmatrix} s_{xN} \\ s_{yN} \\ s_{zN} \end{pmatrix} = \sum_{k=0}^N \left(\mathbf{T}_{start}^{TCP,k} \cdot \begin{pmatrix} v_x \cdot T_{IPO} \\ 0 \\ 0 \end{pmatrix} \right) \quad . \quad (3.28)$$

Die ausschließliche Vorgabe der Bewegung nach Gleichung 3.28 ohne Vorgabe einer Orientierung würde zu einer linearen Bewegung in x -Richtung des Werkzeugkoordinatensystems mit der Geschwindigkeit v_x führen. Die gewünschten Orientierungen sind in \mathbf{S}^* gespeichert. Basierend auf der aktuellen Geschwindigkeit und der Position des Roboters kann in jedem Interpolationsschritt ein Stellwert aus \mathbf{S}^* selektiert werden, sodass die gewünschte Orientierung an der korrekten Position entlang der Kontur eingenommen wird. \mathbf{s}_k^* beschreibt die gewünschte Orientierung des Roboters im Vorlaufpunkt \mathbf{p}_{adv} (siehe Gleichung 3.27). Da \mathbf{s}_k^* im Vorlauf generiert wurde, also in der „Zukunft“ liegt, muss die Verwendung entsprechend verzögert werden. Die Selektion eines Stellwertes basiert auf:

- Verfahrensgeschwindigkeit v_x entlang der Kontur.
- Aktuelle Position \mathbf{r}_{act} des Roboters mit Korrespondenzpunkt \mathbf{p}_{act} auf dem Merkmalspfad \mathbf{P} (Gleichung 3.18).
- Zeitverzögerung $\tau_{sys} = \tau_{rob} + \tau_{fil}$ eines Sollwertes vom Eingang der Robotersteuerung bis zur mechanischen Umsetzung, mit τ_{rob} der Umsetzzeit des Roboters und τ_{fil} der Zeitverzögerung durch Stellwertfilterung.

Mithilfe von v_x und τ_{sys} wird die Distanz \hat{l} geschätzt, die der TCP innerhalb der Verzögerungszeit τ_{sys} entlang \mathbf{P} zurücklegt:

$$\hat{l} = v_x \cdot (\tau_{rob} + \tau_{fil}) = v_x \cdot \tau_{sys} \quad . \quad (3.29)$$

Aus der Prädiktion \hat{l} wird der Punkt $\hat{\mathbf{p}}$ durch Vorlaufen um die Strecke \hat{l} beginnend bei \mathbf{p}_{act} berechnet. Durch diese Schätzung ist immer von der aktuellen Position \mathbf{p}_{act} aus bekannt, an welchem Punkt $\hat{\mathbf{p}}$ sich der Roboter

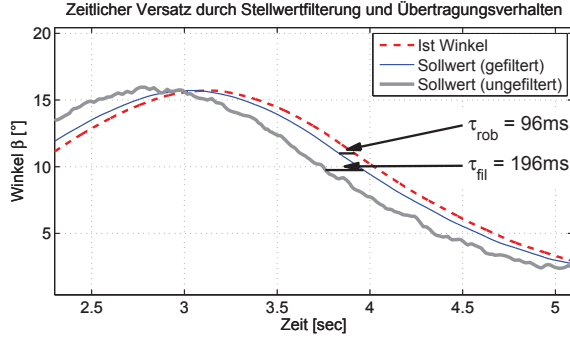


Abbildung 3.13: Zeitverzögerung τ_{fil} , τ_{rob} durch Filterung und Übertragungsverhalten des Roboters.

nach der Systemverzögerungszeit τ_{sys} befinden wird. In jedem Zeitschritt wird daher das sich am nächsten zu $\hat{\mathbf{p}}$ befindende Element

$$\mathbf{s}_j^* = (\mathbf{p}_{adv,j}, s_{\gamma j}, s_{\beta j}, s_{\alpha j})^T \quad \text{mit} \quad \|\mathbf{p}_{adv,j} - \hat{\mathbf{p}}\| = \min \quad (3.30)$$

aus \mathbf{S}^* selektiert, um den vollständigen kartesischen Stellwert

$$\mathbf{s}_N = (s_{xN}, s_{yN}, s_{zN}, s_{\gamma j}, s_{\beta j}, s_{\alpha j})^T. \quad (3.31)$$

zum Zeitpunkt N zu berechnen. Dadurch ist gewährleistet, dass die ermittelte Orientierung nach der Umsetzzeit τ_{sys} an der korrekten Position entlang der Kontur umgesetzt wird.

Die Parameter τ_{rob} und τ_{fil} (Gleichung 3.29) werden ermittelt, indem eine beliebige Trajektorie (mit signifikanten Kurven) an die Robotersteuerung gesendet wird (siehe Abbildung 3.13). Durch Korrelationsanalyse [BSMM01] zwischen dem Sollwert des Roboters (blau) und der tatsächlichen Roboterposition (rot) wird eine Verzögerung des Robotersystems von $\tau_{rob} = 96ms$ ermittelt. Die zusätzliche Stellwertfilterung mittels gleitendem Mittelwertfilter ergibt bei einer Filterlänge von $N_{fil} = 30$ Zyklen bei einer Abtastzeit von $T_{IPO} = 12ms$ eine Verzögerung von $\tau_{fil} = 196ms$.

Durch die Prädiktion ist es trotz der signifikanten Verzögerung durch Stellwertfilterung und durch das Übertragungsverhalten des Roboters möglich, die Kontur mit geringem Fehler zu verfolgen. In dem Experiment in Abbildung 3.14(a) ist die Kante eines stark gewellten Werkstückes

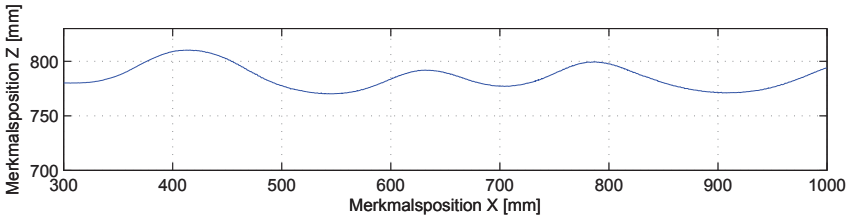
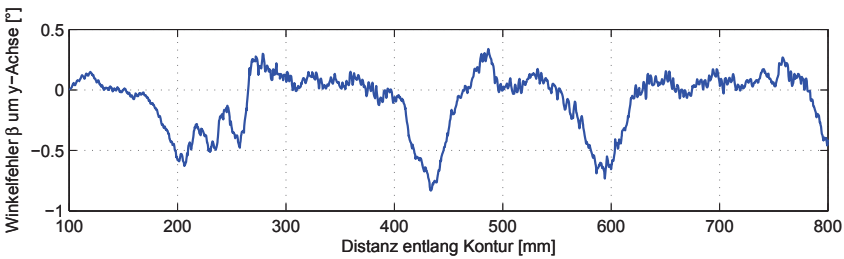
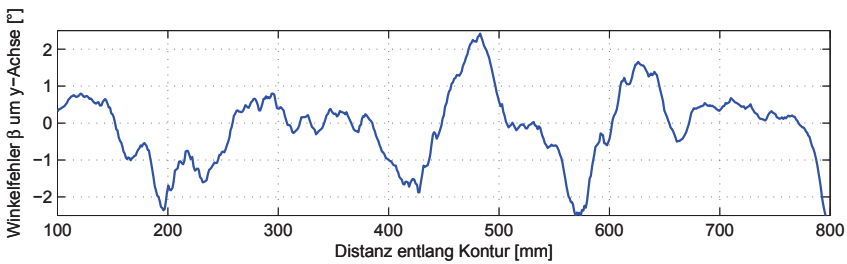
(a) Wellenprofil in der x - z -Ebene(b) Winkelfehler der Rotation um die y -Achse bei linearer Verfahrensgeschwindigkeit von $v_x = 20 \text{ mm/sec}$. Ein Fehler von bis zu $0,8^\circ$ tritt auf.(c) Winkelfehler der Rotation um die y -Achse bei linearer Verfahrensgeschwindigkeit von $v_x = 100 \text{ mm/sec}$. Ein Fehler von bis zu $2,2^\circ$ tritt auf.

Abbildung 3.14: Winkelfehler der Rotation um die y -Achse bei Konturverfolgung an einem Wellenprofil.

in der x - z -Ebene dargestellt. Abbildung 3.14(b) zeigt den Winkelfehler, welcher entsteht, wenn das Werkzeug entlang des Pfades senkrecht zur Kontur orientiert wird. Es entsteht ein Winkelfehler von weniger als 1° bei einer Verfahrgeschwindigkeit von $v_x = 20\text{mm/sec}$. Abbildung 3.14(c) zeigt den Winkelfehler bei einer Verfahrgeschwindigkeit von $v_x = 100\text{mm/sec}$. Dabei erhöht sich der Fehler auf über 2° , da die Stellwerte zur Rauschunterdrückung mit einem gleitenden Mittelwertfilter mit der Filterverzögerungszeit τ_{fil} geglättet werden; somit wird die Glättung bei höherer Geschwindigkeit auf ein längeres Pfadsegment angewendet und Konturen des Pfades werden stärker verschliffen. Zur Reduktion dieses Fehlers durch zeitbasierte Filterung wird in Abschnitt 3.2.4 eine geometrische Filterung vorgestellt. Da bisher nur die Orientierung des Werkzeuges entlang der Kontur betrachtet wurde, wird im folgenden Abschnitt zunächst die translatorische Regelung des Roboters betrachtet, um die Aufgabendefinition nach Tabelle 3.1 (Seite 57) vollständig zu erfüllen.

3.2.3 Berechnung translatorischer Stellwerte

Bei der Berechnung der translatorischen Bahnkorrektur müssen die folgenden Randbedingungen der Aufgabendefinition berücksichtigt werden:

- Die Vorgabe der x -Position in Werkzeugkoordinaten erfolgt durch Vorgabe der Geschwindigkeit v_x entlang der Kontur.
- Die Regelung der Position durch Bildinformationen in y - und z -Richtung des Werkzeugkoordinatensystems müssen einzeln aktiviert bzw. deaktiviert werden können (dies ist z.B. dann notwendig, wenn nur Kraftregelung z.B. in z -Richtung eingesetzt wird).

Ein trivialer Ansatz zur Vorgabe der kartesischen Position aus Bildinformationen wäre die direkte Vorgabe der dreidimensionalen Merkmalsposition \mathbf{p}_{adv} als Stellwert unter Berücksichtigung der aktuellen Geschwindigkeit v_x und des gewünschten Offsets d_y und d_z zur Kontur. Dadurch würde der Roboter jedoch direkt entlang des Merkmalspfades \mathbf{P} geführt werden. Eine Deaktivierung einzelner Richtungen in Werkzeugkoordinaten, also relativ zur aktuellen Orientierung, wäre nicht einfach realisierbar, sodass z.B. eine gleichzeitige Kraftregelung in z -Richtung des Werkzeugkoordinatensystems kaum durchführbar wäre.

Genau betrachtet ist bei Vorgabe einer Geschwindigkeit v_x in x -Richtung des Werkzeugkoordinatensystems und einer **idealen** Vorgabe der Orientie-

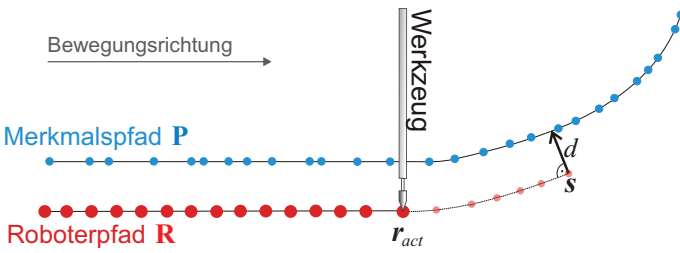


Abbildung 3.15: Durch die Umsetzzeit τ_{rob} des Roboters ist die zukünftige Roboterpose anhand des aktuellen Stellwertes s vorausschauend bekannt. Da auch die Bildinformation vorausschauend bekannt ist, kann der Abstand d_y und d_z im Koordinatensystem der zukünftigen Roboterpose s ermittelt werden.

nung nach Gleichung 3.31 keine Regelung der Position in y - und z -Richtung notwendig. Durch Ungenauigkeiten der Orientierung entfernt sich jedoch der TCP während der Bewegung von der Kontur. Die Positionsvorgabe wird daher lediglich zur notwendigen Korrektur dieser Ungenauigkeiten benötigt. Im Gegensatz zur Vorgabe der absoluten Orientierung (Gleichung 3.26) ist der Prädiktionshorizont der Positionskorrektur deutlich kürzer, da tatsächliche Fehler des Abstandes d_y und d_z zwischen den Merkmalen und der aktuellen Roboterposition \mathbf{r}_{act} korrigiert werden müssen. Wie zuvor beschrieben, wird ein an den Roboter gesendeter Stellwert s mit der Zeitverzögerung τ_{rob} umgesetzt. D.h., s bietet eine gute Schätzung der zukünftigen Roboterpose, welche nach der Umsetzzeit τ_{rob} eingenommen wird. Da auch die Bildmerkmale vorausschauend ermittelt werden, kann der Abstand in y - und z -Richtung bereits für die Zeit τ_{rob} vorausschauend im Koordinatensystem der zukünftigen Roboterpose s ermittelt werden. Abbildung 3.15 zeigt das Messprinzip; Gleichung 3.28 erweitert sich zu

$$\begin{pmatrix} s_{xN} \\ s_{yN} \\ s_{zN} \end{pmatrix} = \sum_{k=0}^N \left(\mathbf{T}_{start}^{TCP,k} \cdot \begin{pmatrix} v_x \cdot T_{IPO} \\ 0 \\ 0 \end{pmatrix} + \mathbf{T}_{start}^{TCP,k} \cdot \begin{pmatrix} 0 \\ A_y \cdot d_y \\ A_z \cdot d_z \end{pmatrix} \right) \quad , \quad (3.32)$$

mit $A_y = 0$, $A_z = 0$ zur Deaktivierung der Regelung in der entsprechenden Richtung bzw. $A_y = 1$, $A_z = 1$ zur Aktivierung der Regelung in der entsprechenden Richtung. Mit dieser Gleichung wird der Abstand in y - und in z -Richtung zur Kontur beeinflusst. Messergebnisse zur translatorischen

Vorgabe der Position werden im nachfolgenden Abschnitt im Zusammenhang mit der geometrischen Filterung der Messwerte gezeigt.

3.2.4 Filterung des Merkmalspfades

In den vorangegangenen Abschnitten wurde die Generierung der rotatorischen Korrektur durch Approximation einer Tangente an die Kontur gezeigt. Nach Gleichung 3.19 wird diese Tangente durch eine Sekante genähert. Einerseits ist die Näherung durch eine Sekante sehr einfach zu berechnen, andererseits ist durch die Berechnung aus lediglich zwei Stützstellen eine Empfindlichkeit gegenüber Messrauschen zu erwarten. Weiterhin gehen Konturinformationen durch die stückweise Linearisierung verloren. Die nachfolgenden Experimente zeigen, dass durch die Empfindlichkeit gegenüber Messrauschen eine starke Glättung der Messwerte notwendig ist, so dass durch Filterung weitere Konturinformationen verloren gehen können. Nachfolgend wird alternativ die lineare und quadratische Pfadapproximation in \mathbb{R}^3 vorgestellt und anschließend der Einfluss auf die Konturverfolgung untersucht.

Dreidimensionale lineare Regression

Zur Approximation einer Tangente an reale Messwerte müssen benachbarte Punkte des gesuchten Tangentenpunktes mit berücksichtigt werden. Je kleiner die Umgebung gewählt wird, desto empfindlicher reagiert das Ergebnis auf Messrauschen; eine große Umgebung kann die ermittelte Orientierung stark verfälschen. Die Tangente im Punkt \mathbf{p}_{adv} nach Gleichung 3.19 wurde durch die Nachbarpunkte \mathbf{p}_Q und \mathbf{p}_R approximiert. Für die nachfolgend beschriebene lineare Regression werden alle Elemente zwischen \mathbf{p}_Q und \mathbf{p}_R zur Approximation verwendet. D.h., die lineare Regression wird in jedem Interpolationsschritt auf ein Segment

$$\mathbf{p}_j \quad \text{mit } j = Q, \dots, R \quad (3.33)$$

in der Umgebung des Vorlaufpunktes \mathbf{p}_{adv} angewendet. Die Merkmale \mathbf{p}_j werden auf eine Regressionsebene

$$\vec{E}_R : \quad y = a + b \cdot x + c \cdot z \quad (3.34)$$

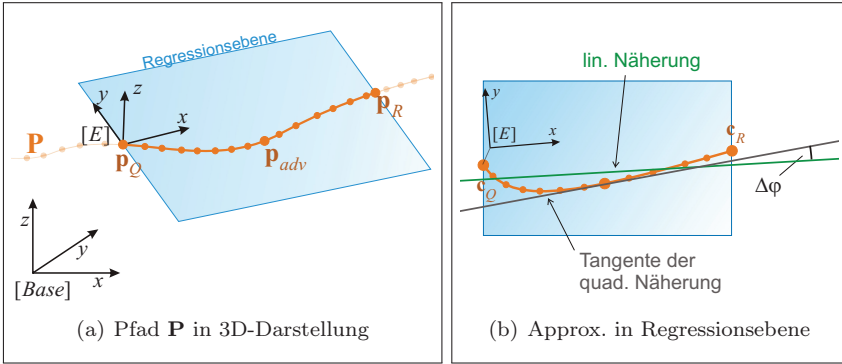


Abbildung 3.16: Berechnung einer Tangente durch lineare bzw. quadratische Pfadapproximation in \mathbb{R}^3 nach Projektion des Pfades P auf eine Regressionsebene. Lineare Approximation erzeugt den Winkelfehler $\Delta\varphi$.

projiziert (siehe Abbildung 3.16). Die Berechnung der Regressionsebene durch die Punkte \mathbf{p}_j wird in Anhang A gezeigt. Durch die Projektion von \mathbf{p}_j auf die Ebene wird das projizierte Pfadsegment

$$\mathbf{e}_j \quad \text{mit } j = Q, \dots, R \quad (3.35)$$

berechnet (die Berechnung der Regressionsebene und die Projektion sind in Anhang A dargestellt). Das Koordinatensystem der Regressionsebene $K_{[E]}$ wird in Abhängigkeit des Normalenvektors \vec{n}_R der Ebene im Basissystem folgendermaßen definiert:

$$\begin{aligned} \vec{a}_E &= \frac{\vec{n}_R}{\|\vec{n}_R\|}; \quad \vec{n}_E = \frac{\mathbf{e}_R - \mathbf{e}_Q}{\|\mathbf{e}_R - \mathbf{e}_Q\|}; \quad \vec{o}_E = \vec{a}_E \times \vec{n}_E \\ \mathbf{T}_{base}^E &= \begin{pmatrix} \vec{n}_E & \vec{o}_E & \vec{a}_E & \mathbf{c}_1 \\ 0 & 0 & 0 & 1 \end{pmatrix} \end{aligned} \quad (3.36)$$

Um \mathbf{e}_j in \mathbb{R}^2 zu beschreiben wird \mathbf{e}_j in das Ebenenkoordinatensystem $K_{[E]}$ transformiert:

$$\mathbf{c}_j = (\mathbf{T}_{base}^E)^{-1} \cdot \begin{pmatrix} \mathbf{e}_j \\ 1 \end{pmatrix}. \quad (3.37)$$

Die z -Komponente von $\mathbf{c}_j = (c_{jx}, c_{jy}, 0)^T$ ist Null, da die z -Achse des Ebenenkoordinatensystems durch den Normalenvektor von \vec{E}_R definiert

ist. Die Regressionsgerade mit den Parametern a_g und b_g in \mathbb{R}^2 kann in Abhängigkeit der Koordinaten u und v aus c_{jx} und c_{jy} berechnet werden (siehe Anhang A):

$$v = a_g + b_g \cdot u \quad (3.38)$$

Aus Gleichung 3.38 wird ein Punkt $(u_1, v_1)^T$ in der Mitte des betrachteten Segmentes und ein beliebiger weiterer Punkt $(u_2, v_2)^T$ entlang der Regressionsgerade berechnet. Diese Punkte werden in Basiskoordinaten transformiert und zur Berechnung eines Vektors \vec{t}_{adv} zur Beschreibung der Tangente im Vorlauffpunkt \mathbf{p}_{adv} verwendet:

$$\mathbf{p}_{adv} = \mathbf{T}_{base}^E \cdot \begin{pmatrix} u_1 \\ v_1 \\ 0 \\ 1 \end{pmatrix}; \quad \mathbf{p}_{tan} = \mathbf{T}_{base}^E \cdot \begin{pmatrix} u_2 \\ v_2 \\ 0 \\ 1 \end{pmatrix} \quad (3.39)$$

$$\vec{t}_{adv} = \mathbf{p}_{tan} - \mathbf{p}_{adv} \quad (3.40)$$

Berechnen der Tangente mittels genäherter quadratischer Regression im Raum

Abbildung 3.16 zeigt, dass die lineare Regression den Winkelfehler $\Delta\varphi$ gegenüber einer Tangentenbildung aus quadratischer Regression erzeugen kann. Eine Regression zweiter Ordnung im Raum ist ein mathematisch aufwändiges Problem und lässt sich z.B. als eine Optimierungsaufgabe formulieren. Eine zufriedenstellende Lösung ist jedoch die quadratische Regression in der Regressionsebene nach Gleichung 3.34. Die in die Ebene transformierten Punkte $\mathbf{c}_j = (c_{jx}, c_{jy}, 0)^T$ sollen durch die Regressionskurve mit den Koordinaten u und v approximiert werden:

$$v = a_q + b_q \cdot u + c_q \cdot u^2 \quad (3.41)$$

Die unbekannten Parameter a_q, b_q, c_q werden mithilfe der minimalen Fehlerquadratmethode aus den Messwerten \mathbf{c}_j ermittelt (siehe Anhang B). Die Richtung der Tangente an die quadratische Regression wird mittels der Ableitung in der Mitte der Regressionskurve berechnet:

$$v' = b_q + 2 \cdot c_q \cdot u \quad (3.42)$$

Aus Gleichung 3.41 und Gleichung 3.42 wird ein Punkt $(u_1, v_1)^T$ in der Mitte der Regressionskurve ermittelt, und ein Punkt $(u_2, v_2)^T$ aus der Ableitung der Regressionskurve. Entsprechend Gleichung 3.39 und 3.40 wird aus diesen beiden Punkten eine Tangente \vec{t}_{adv} im gefilterten Vorlaufpunkt \mathbf{p}_{adv} berechnet.

Diese Art der Berechnung stellt keine „echte“ Regressionskurve im Raum dar, da diese nur eine Krümmung in einer Dimension aufweist. Da die Regression jedoch immer nur stückweise auf ein kurzes Element p_j mit $j = Q, \dots, R$ des Pfades berechnet wird, beschreibt dies eine ausreichend genaue Approximation. Zudem wird durch die Regressionsebene in der Projektion der Daten immer die stärkste Krümmung des Segmentes dargestellt.

Einfluss der Pfadapproximation auf die Konturverfolgung

Es wurde die Näherung der Tangente an die Kontur im Vorlaufpunkt \mathbf{p}_{adv} auf drei Arten gezeigt:

1. Bildung einer **Sekante** durch zwei Punkte \mathbf{p}_Q und \mathbf{p}_R in der Umgebung von \mathbf{p}_{adv} .
2. **Lineare Regression** durch eine Punktwolke in der Umgebung von \mathbf{p}_{adv} zwischen \mathbf{p}_Q und \mathbf{p}_R .
3. **Quadratische Regression** durch eine Punktwolke in der Umgebung von \mathbf{p}_{adv} zwischen \mathbf{p}_Q und \mathbf{p}_R .

Der Einfluss dieser drei Approximationen auf die Konturverfolgung wird nun verglichen. Aus der approximierten Tangente \vec{t}_{adv} im Vorlauf wird die gewünschte Orientierung des Werkzeuges berechnet (siehe Gleichung 3.22). Die daraus resultierende Bewegung des Roboters wird in einem Experiment untersucht, wobei die Winkelgeschwindigkeit der z -Achse an einer leicht gekrümmten Kontur beurteilt wird (siehe Abbildung 3.17). Störungen entstehen in der Winkelgeschwindigkeit, wenn die Orientierung der Kontur aus einer Sekante durch nur zwei Konturpunkte \mathbf{p}_Q und \mathbf{p}_R ermittelt wird (rote Kurve). Wird eine Regression zwischen \mathbf{p}_Q und \mathbf{p}_R berechnet (blaue Kurve), werden deutlich kleinere Störungen erzeugt.

Bei einer Sekantenlänge von z.B. $l_{sek} = 20mm$, erzeugt ein Rauschen von $\Delta y = 0.5mm$ in einem der beiden Sekantenpunkte die Änderung

$$\Delta\alpha = \arctan\left(\frac{\Delta y}{l_{sek}}\right) = 1.43^\circ \quad (3.43)$$

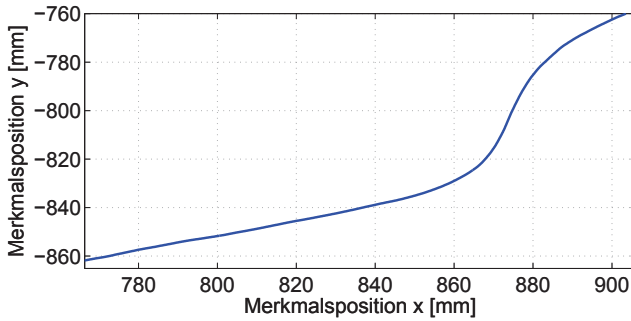
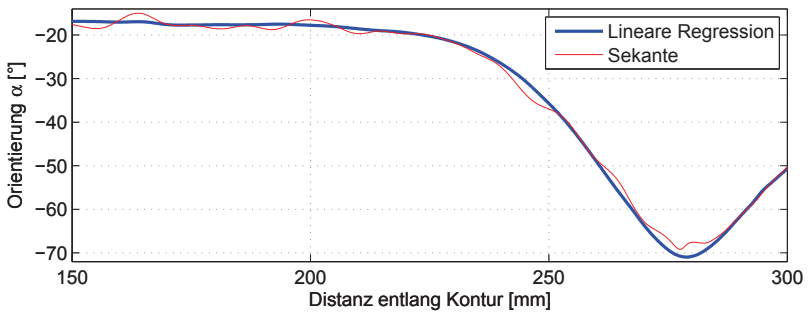
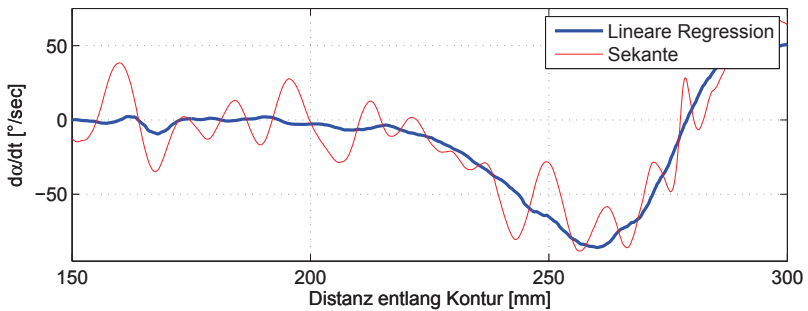
(a) Kontur in der x - y Ebene.(b) Rotation α um die z -Achse.(c) Rotationsgeschwindigkeit $d\alpha/dt$ um die z -Achse.

Abbildung 3.17: Winkel und Winkelgeschwindigkeit bei der Anpassung der Orientierung und Position des Werkzeuges an eine leicht gekrümmte Kontur bei einer Geschwindigkeit von $v_x = 60 \text{ mm/sec}$.

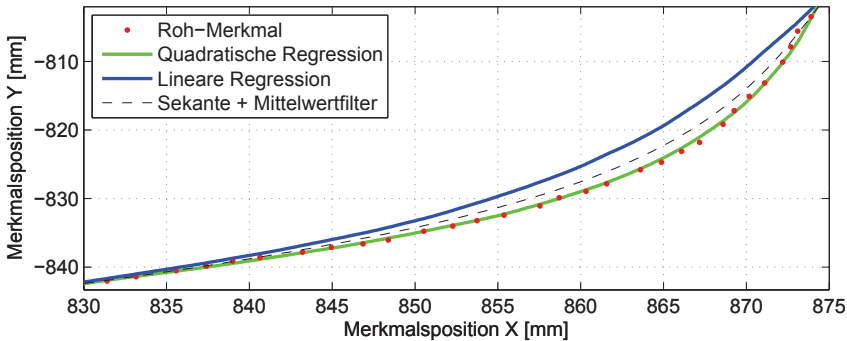
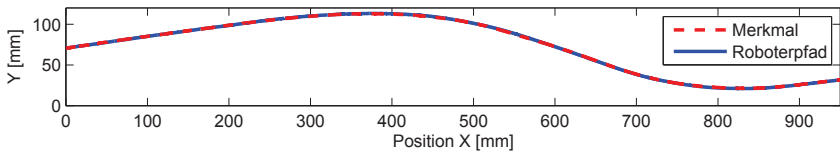


Abbildung 3.18: Filterung der Merkmalsposition durch quadratische Regression, lineare Regression und Mittelwertfilterung.

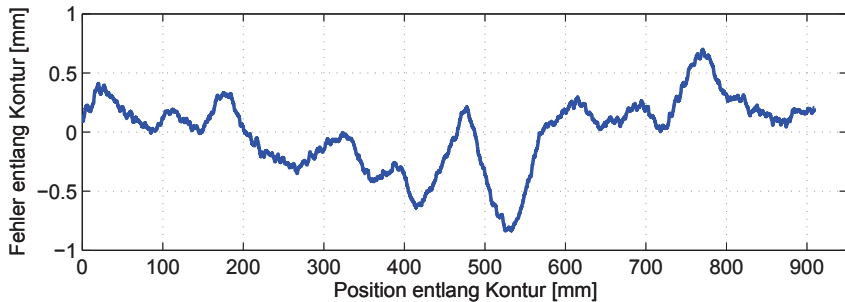
in der Orientierung der Tangente. D.h., durch Messrauschen wird das Werkzeug (und somit das Kamerasystem) in Schwingungen versetzt. Die Unruhe im System begünstigt verstärktes Messrauschen, sodass Schwingungen während des gesamten Prozesses auftreten. Wird die Tangente durch Regression approximiert (blaue Kurve) ist das Verhalten deutlich ruhiger, da der Einfluss von einzelnen Messfehlern und Messrauschen reduziert wird.

Weiterhin wird der Vorteil der quadratischen Regression gegenüber der linearen Regression in Abbildung 3.18 gezeigt. Insbesondere die translatorische Regelung des Abstandes zwischen Werkzeug und Kontur wird verbessert, wenn die gemessenen Merkmale entlang der Kontur (rote Punkte) durch quadratische Regression geglättet werden (grün). Um die Störungen bei Ermittlung der Orientierung durch eine Sekante zu reduzieren, müssen die Stellwerte stark gefiltert werden. Ein gleitender Mittelwertfilter verfälscht die Kontur (gestrichelte Kurve). Auch die Wahl des Merkmals aus dem Mittelpunkt der linearen Regression verfälscht die Kontur signifikant (blaue Kurve). Die quadratische Regression erzeugt die geringste Verfälschung der Kontur.

Auch bei Konturen, die nicht insgesamt durch eine quadratische Funktion darstellbar sind, liefert die quadratische Approximation eine hohe Bahn-treue, da pro Zeitschritt jeweils nur ein kurzes Pfadsegment von z.B. 20mm betrachtet wird, welches wiederum ausreichend genau durch eine quadratische Funktion repräsentiert werden kann. Es wird daher für alle Experi-



(a) Zu verfolgende Kontur und der entsprechende Pfad des Roboters.



(b) Positionsfehler des Roboters entlang der Kontur.

Abbildung 3.19: Anpassung der Position entlang einer leicht gebogenen Kontur bei einer Geschwindigkeit von $v_x = 600\text{mm/sec}$. Eine Abweichung von weniger als 1mm tritt entlang des Pfades auf.

mente die quadratische Regression zur Glättung der Kontur verwendet. In Abbildung 3.19 ist eine Kontur dargestellt, welche mit $v_x = 600\text{mm/sec}$ verfolgt werden soll. Die Orientierung und die Position des Roboters werden angepasst. Um den absoluten translatorischen Fehler zu beurteilen, wurde ein Stift als Roboterwerkzeug montiert, sodass durch die Verfolgung der Kontur diese nachgezeichnet wird. Es ergibt sich entlang der unbekannten Kontur trotz der hohen Geschwindigkeit ein Fehler von unter 1mm entlang der gesamten Kontur.

3.2.5 Automatische Anpassung der Verfahrensgeschwindigkeit v_x

Einige Applikationen erfordern die Begrenzung der Winkelgeschwindigkeit des Werkzeuges, um z.B. Fliehkräfte bei starken Krümmungen zu mini-

mieren. Darüber hinaus werden hohe Gelenkgeschwindigkeiten vermieden, wenn bei starken Krümmungen entlang der Kontur die Verfahrensgeschwindigkeit der Kurve automatisch angepasst wird.

Die reine Begrenzung der Winkelgeschwindigkeit würde zu einer Fehlstellung der Winkel entlang der Kontur führen, wenn die Verfahrensgeschwindigkeit v_x dabei weiter unverändert bliebe. Die Begrenzung muss also über die Begrenzung der Geschwindigkeit v_x realisiert werden. Zudem muss das Erreichen der Geschwindigkeitsgrenze bereits vorausschauend erkannt werden, um eine angemessene Bremsbeschleunigung berücksichtigen zu können. Abhängig von der aktuellen Geschwindigkeit v_x beträgt die zurückgelegte Distanz s_x entlang der Kontur pro Interpolationsschritt k mit dem Interpolationstakt $T_{IPO} = 4ms$ des Roboters

$$s_{x,k} = v_{x,k} \cdot T_{IPO} \quad . \quad (3.44)$$

Die Ableitung der kartesischen Winkel α, β, γ nach der zurückgelegten Distanz $s_{x,k}$ am Punkt $\mathbf{p}_{adv,k}$ wird aus Gleichung 3.27 vorausschauend berechnet. Beispielhaft wird die Änderung des Winkels pro Strecke für den Winkel α um die z -Achse berechnet:

$$\frac{d\alpha_k}{ds_{x,k}} \approx \frac{\alpha_k - \alpha_{k-1}}{\|\mathbf{p}_{adv,k} - \mathbf{p}_{adv,k-1}\|} \quad . \quad (3.45)$$

Unter Vorgabe einer maximal erlaubten Winkelgeschwindigkeit $\frac{d\alpha}{dt}$ wird die maximal mögliche Geschwindigkeit $v_{xmax,k}$ berechnet:

$$v_{xmax,k} = \frac{ds_{x,k}}{dt} = \frac{ds_{x,k}}{d\alpha_k} \cdot \frac{d\alpha}{dt} = \frac{\frac{d\alpha}{dt}}{\frac{d\alpha_k}{ds_{x,k}}} \quad . \quad (3.46)$$

Die maximale Verfahrensgeschwindigkeit $v_{xmax,k}$ wird für den Vorlaufpunkt $\mathbf{p}_{adv,k}$ (im Abstand l_{adv}) berechnet. Somit muss die Geschwindigkeit nicht unmittelbar, sondern erst nach der Vorlaufstrecke erreicht werden. Die Beschleunigung a und die Vorlaufzeit Δt ergeben sich abhängig von der Vorlaufstrecke l_{adv} , der aktuellen Geschwindigkeit v_x und der maximalen Geschwindigkeit $v_{x,max}$:

$$l_{adv} = \frac{1}{2} \cdot a \cdot (\Delta t)^2 + v_x \cdot \Delta t \quad \text{mit } a = \frac{v_x - v_{xmax}}{\Delta t} \quad (3.47)$$

$$\Rightarrow \Delta t = \frac{l_{adv}}{\frac{3}{2}v_x - \frac{1}{2}v_{xmax}}$$

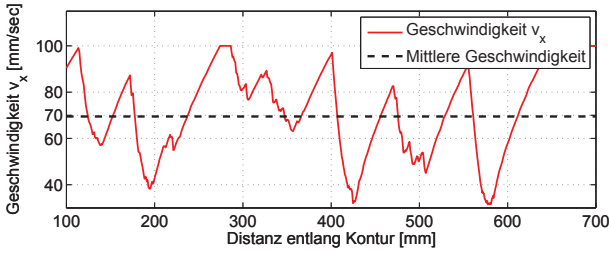
Abbildung 3.20(a) zeigt das Geschwindigkeitsprofil bei automatischer Begrenzung der Winkelgeschwindigkeit auf $d\beta/dt \leq 40^\circ/sec$ an einem Wellenprofil (rot). Es wird eine mittlere Geschwindigkeit \bar{v}_x von ca. $70mm/sec$ erreicht (schwarz gestrichelt). Abbildung 3.20(b) zeigt den Winkelfehler entlang der Kontur. Verglichen mit konstanter Geschwindigkeit von $v_x = 70mm/sec$ (schwarz) ist der Winkelfehler entlang der Kontur bei automatischer Begrenzung der Geschwindigkeit geringer (rot). Abbildung 3.20(c) zeigt die Winkelgeschwindigkeit um die y -Achse. Die vorgegebene Begrenzung der Winkelgeschwindigkeit wird nahezu vollständig eingehalten (rot), da die Verfahrgeschwindigkeit nach Gleichung 3.46 entsprechend der maximalen Winkelgeschwindigkeit reduziert vorgegeben wird.

3.2.6 Erkennung von Ecken

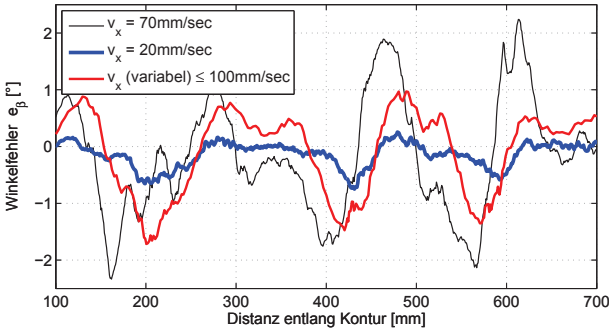
Entlang kontinuierlicher Konturverläufe glättet die vorgestellte quadratische Regression den Pfad mit hoher Bahntreue. Die automatische Anpassung der Geschwindigkeit vermeidet schnelle Orientierungsänderungen und somit Fliehkräfte. Bei Erreichen einer Ecke genügen diese Mechanismen jedoch nicht mehr. Eine Ecke im Merkmalspfad wird durch die Filterung stark verschliffen. Abbildung 3.21(a) zeigt eine Ecke vor der Filterung (blau) und nach Filterung (rot). Abbildung 3.21(b) zeigt die im Vorlauf berechnete Änderung des Winkels α um die z -Achse aus dem gefilterten Pfad. An einer Ecke müsste der Winkel sprungförmig verändert werden, d.h., es wäre eine unendlich hohe Winkelgeschwindigkeit gefordert. Durch die Filterung des Pfades ergibt sich im Bereich der Ecke (Punkt A) ein lokales Maximum der Winkeländerung pro Strecke von ca. $8^\circ/mm$. Anhand des Maximums wird eine starke Krümmung festgestellt, jedoch nicht die exakte Position der Ecke. Um die exakte Position der Ecke zu bestimmen, muss diese in den ungefilterten Daten gesucht werden. Die Position einer möglichen Ecke im Bereich der starken Krümmung wird mit dem folgenden iterativen Algorithmus³⁾ bestimmt (siehe auch Abbildung 3.22):

1. Um den Punkt A herum wird ein Dreieck mit den Eckpunkten A , B , C erstellt mit $\|AB\| = \|AC\| = s$. Bei der ersten Iteration wird mit einer Seitenlänge von z.B. $s = 12mm$ gestartet.

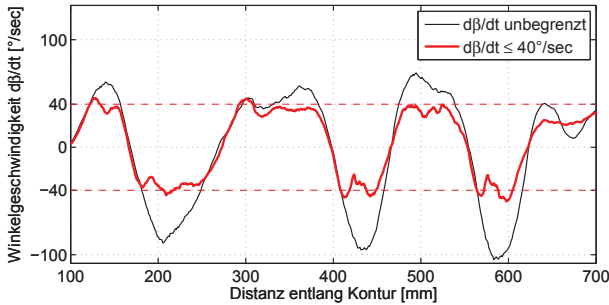
³⁾Der Algorithmus ist eine Modifikation des „Curvature Scale Space“ nach Mokhtarian und Suomela [MS98]. Dabei handelt es sich um eine 2D-Eckenerkennung mit variabler Suchgröße zur präzisen Bestimmung von Ecken.



(a) Geschwindigkeit $v_{x,i}$ bei Limitierung $d\beta/dt \leq 40^\circ/\text{sec}$.

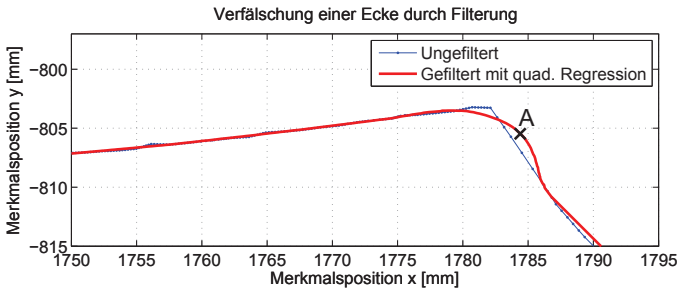


(b) Winkelfehler β bei unterschiedlichen Geschwindigkeiten.

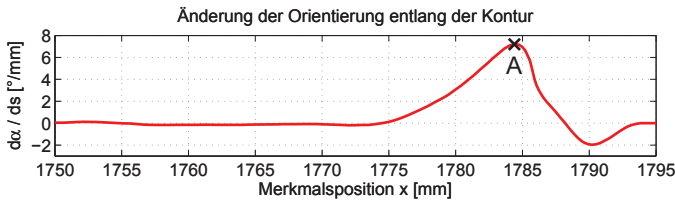


(c) Winkelgeschwindigkeit β .

Abbildung 3.20: Automatische Geschwindigkeitskorrektur am Wellenprofil. Die Winkelgeschwindigkeit $d\beta/dt \leq 40^\circ/\text{sec}$ um die y -Achse wird durch automatische Begrenzung der Geschwindigkeit v_x eingehalten.



(a) Die Ecke wird durch Filterung des Pfades stark verschliffen.



(b) Im Vorlauf berechnete Änderung des Winkels α um die z -Achse entlang des gefilterten Merkmalspfades.

Abbildung 3.21: Die Ecke wird durch Filterung stark verschliffen. Die exakte Position der Ecke kann nicht anhand der maximalen Krümmung (Punkt A) aus den gefilterten Daten ermittelt werden.

2. Der Punkt A wird zwischen den Punkten B und C verschoben (ohne B und C zu verändern, siehe Abbildung 3.22), mit $\|AB\|, \|AC\| > \frac{s}{2}$, bis der Winkel ϕ im Eckpunkt A zwischen AB und AC ein Maximum erreicht.
3. Wenn der maximale Winkel ϕ_{max} unterhalb eines Schwellwertes von z.B. 50° liegt, wird diese mögliche Ecke verworfen. Ist ϕ_{max} oberhalb des Schwellwertes, wird die Seitenlänge s um $3mm$ verringert und bei Schritt 1 fortgefahren, wobei der Mittelpunkt A des Dreieckes nun auf den zu ϕ_{max} korrespondierenden Punkt entlang der Kontur gelegt wird. Dieser Algorithmus wird wiederholt, solange bis $s \leq 3mm$ ist.

Durch die iterative Verkleinerung der Dreiecksgröße wird die Position der Ecke exakt bestimmt. Durch den Start des Algorithmus mit relativ großen Dreiecksseiten s wird die Empfindlichkeit gegenüber Messrauschen vermin-

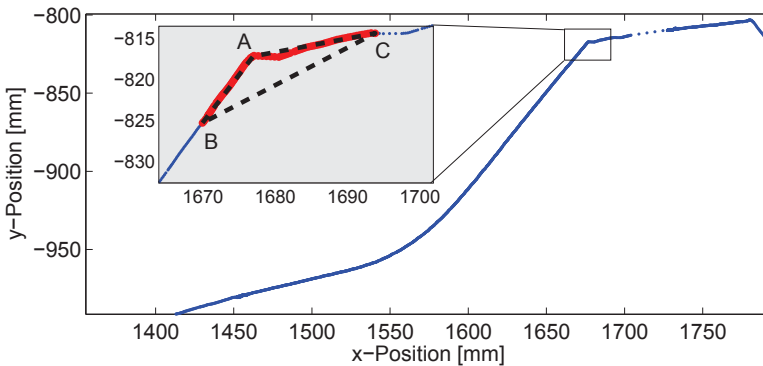


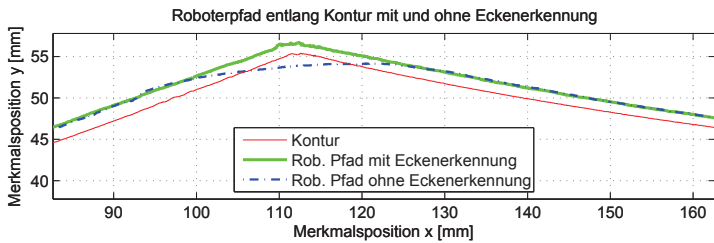
Abbildung 3.22: Einpassung eines Dreieckes an die 3D-Kontur zur Ermittlung einer Ecke.

dert. Die Einpassung eines kleinen Dreieckes sorgt für eine exakte Bestimmung des Eckpunktes.

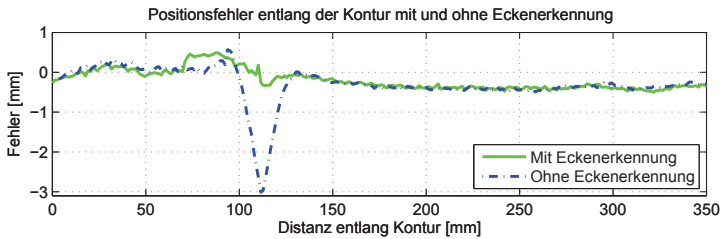
Bei Erreichen einer Ecke verhält sich das System folgendermaßen:

1. Bis zum Erreichen des Eckpunktes A werden lediglich Daten aus \mathbf{P} bis zu diesem Eckpunkt verarbeitet.
2. Die Geschwindigkeit v_x des Werkzeuges wird bis zum Eckpunkt A auf $v_x = 0$ reduziert.
3. Im Eckpunkt wird die Orientierung des Werkzeuges an die Kontur hinter der Ecke angepasst.
4. Die Konturverfolgung wird hinter der Ecke normal weiter durchgeführt.

In Abbildung 3.23 ist der Roboterpfad entlang einer Ecke gezeigt, wobei zur besseren Darstellung ein gewünschter Offset zwischen Werkzeug und Kontur eingehalten wird. Ohne Berücksichtigung der Ecke entlang der Kontur wird diese stark verschliffen, sodass ein Fehler entlang der Kontur von bis zu 3mm entsteht. Bei Berücksichtigung der Ecke reduziert sich der Fehler auf weniger als $0,5\text{mm}$.



(a) Roboterpfad entlang einer Kontur mit Ecke



(b) Konturfehler mit und ohne Berücksichtigung der Ecke

Abbildung 3.23: Roboterpfad entlang einer Kontur mit Ecke. Der Fehler reduziert sich in diesem Experiment von 3mm auf weniger als $0,5\text{mm}$ durch Berücksichtigung der Ecke. Zur besseren Darstellung wurde mit einem gewünschten Offset $d_y = 3\text{mm}$ zur Kontur verfahren.

3.3 Zusammenfassung

In diesem Kapitel wurden Algorithmen zur bildgestützten Verfolgung einer sichtbaren Kontur vorgestellt. Die beliebig geformte Kontur, an welche die Bewegung des Roboters in allen sechs Freiheitsgraden angepasst wurde, ist als komplett unbekannt angenommen worden. Durch das Messprinzip existiert ein der Vorlauftlänge der Kamera entsprechender Prädiktionshorizont, in dem die Form der Kontur vor Erreichen des TCP erfasst ist. Bedingt durch das Messprinzip kann es zu unsortierten Messwerten entlang der Pfadrichtung der Kontur kommen. Dieses Problem wurde durch die geometrische Sortierung der Messwerte behandelt. Zudem wurde die Verzögerungszeit durch Bildaufnahme und Bildverarbeitung untersucht und in der Messwertverarbeitung entsprechend berücksichtigt. Aus den Messwerten wurden Stellwerte für das Robotersystem generiert, wobei die Verzögerung des Systems durch Filterung und die Umsetzzeit des Roboters bei der Selektion der Stellwerte mit einbezogen wurden. Da gerade bei einer am Roboter montierten Kamera mit Messrauschen zu rechnen ist, wurden verschiedene Filtermechanismen untersucht, unter den Randbedingungen der Bahntreue und mit der Voraussetzung, diese Filterung innerhalb des Interpolationstaktes von $4ms$ berechnen zu können. Dabei hat sich die stückweise quadratische Regression, welche in einer Ausgleichsebene eines Pfadsegmentes berechnet wird, als vorteilhaft erwiesen. Weiterhin wurde die Verfahrensgeschwindigkeit des Systems in Kurven automatisch angepasst sowie Ecken im Verlauf erkannt und berücksichtigt.

Zu den einzelnen Komponenten wurden Experimente an unterschiedlichen Konturformen gezeigt. Es hat sich herausgestellt, dass man bei leicht gekrümmten Konturen eine Verfahrensgeschwindigkeit von $v_x = 600mm/sec$ erreichen kann, wobei ein Positionsfehler von $1mm$ unterschritten wird. Für stärker gekrümmte Konturen hat sich die automatische Anpassung der Verfahrensgeschwindigkeit bewährt, sodass Winkelfehler von weniger als 2° erreicht werden können. Ebenso hat die Erkennung von Ecken dazu geführt, dass auch an nicht gleichmäßig gekrümmten Verläufen eine hohe Bahntreue des Roboters zu der Werkstückkontur möglich ist.

4 Kraftregelung

Tritt ein robotergeführtes Werkzeug in Kontakt mit der Umgebung, dann erfordert dies eine Regelung der Kontaktkraft, um Schäden an Werkzeug und Werkstück zu vermeiden. Bei Kontaktaufgaben, wie z.B. dem robotergeführten Schleifen, Polieren oder Entgraten sind darüber hinaus prozessspezifische Kontaktkräfte einzuhalten, welche direkten Einfluss auf die Qualität des Bearbeitungsprozesses haben. Zur Regelung der Kontaktkraft müssen die aktuell auftretenden Kräfte gemessen werden. Üblicherweise wird dazu ein Kraft-Momenten Sensor (KMS) zwischen den Handflansch des Roboters und das Werkzeug montiert (siehe Abbildung 4.1). Der KMS misst alle Kräfte und Momente, welche auf das Messsystem wirken – also auch Gravitations-, Trägheits- und Zentrifugalkräfte. Diese Anteile zählen nicht zur Kontaktkraft und müssen kompensiert werden, indem sie berechnet und vom Messwert subtrahiert werden (zur Vereinfachung des Sprachgebrauches sind *Kräfte und Momente* gemeint, wenn nachfolgend allgemein *Kräfte* erwähnt werden).

Trägheitskräfte treten besonders bei sensorgeführten Bewegungen auf, da eine z.B. durch Visual Servoing kommandierte Beschleunigung in Trägheitskräften resultiert. Der Fokus dieses Kapitels liegt auf der Kompensa-

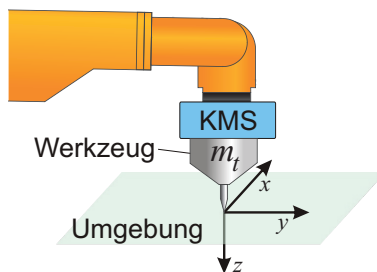


Abbildung 4.1: Anbringung des Kraft-Momenten Sensors (KMS) am Roboter. Das am KMS montierte Werkzeug hat die Masse m_t .

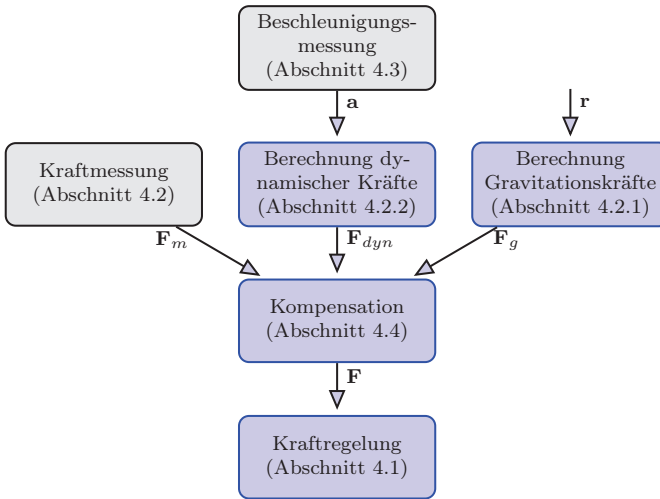


Abbildung 4.2: Übersicht mit Kapitelangabe. Die statische Kraft \mathbf{F}_g (Gravitation) wird aus der Roboterpose \mathbf{r} , die dynamische Kraft \mathbf{F}_{dyn} (Trägheit) wird aus der Beschleunigung \mathbf{a} berechnet und zur Kompensation der Kraftmessung \mathbf{F}_m verwendet, um die Kontaktkraft \mathbf{F} zu berechnen.

tion dieser Kräfte, um die Kontaktkraft zu ermitteln. Nach einer kurzen Einführung über den in dieser Arbeit verwendeten Kraftregelkreis wird die Kraftkompensation vorgestellt, und verschiedene Methoden zur Berechnung der Trägheitskräfte miteinander verglichen. Eine Übersicht des Kapitels wird in Abbildung 4.2 gegeben. Teile dieses Kapitels wurden in [KKWSK10] veröffentlicht. Es kommen folgende Sensoren zum Einsatz:

- 6D KMS: Steifer Kraft-Momenten Sensor zur Messung der Kräfte und Momente in allen sechs Raumrichtungen.
- 12D KMS: Nachgiebiger KMS mit zusätzlichen Beschleunigungssensoren zur Messung der Kräfte und Momente sowie translatorischer und rotatorischer Beschleunigungen in allen sechs Raumrichtungen.
- Drei 2-Achs Beschleunigungssensoren in freier Anordnung zur Messung der translatorischen und rotatorischen Beschleunigungen in allen sechs Raumrichtungen.

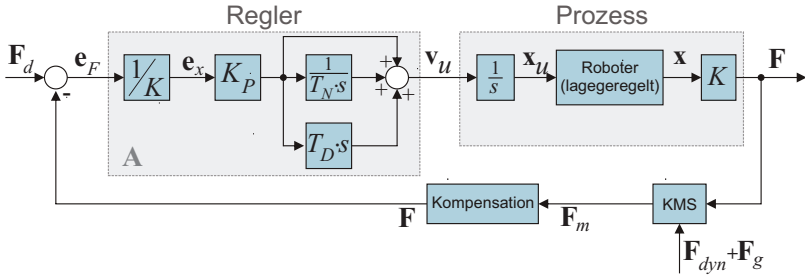


Abbildung 4.3: Positionsbasierte explizite Kraftregelung eines lagegeregelten Roboters mit PID-Regler. Die Steifigkeit K der Umgebung wird berücksichtigt. Zur Ermittlung der Kontaktkraft \mathbf{F} müssen Gravitations- und Trägheitskräfte im Kraftmesswert \mathbf{F}_m kompensiert werden.

4.1 Positionsbasierte Kraftregelung

Die Einhaltung der gewünschten Kontaktkraft \mathbf{F}_d wird auf Basis einer *positionsbasierten expliziten Kraftregelung* mit PID-Regler implementiert. Der Regelkreis ist in Abbildung 4.3 dargestellt. Diese Art der Regelung wird allgemein auch als Admittanzregelung bezeichnet [ZH97]. Die kartesischen Achrichtungen der Kraftregelung werden als voneinander unabhängig betrachtet (daraus folgen ausschließlich Werte auf der Hauptdiagonalen der Admittanzmatrix \mathbf{A}). Die Regelung wird nachfolgend beispielhaft für die z -Richtung vorgestellt. Die Admittanz ordnet einer Kraft e_{F_z} eine Geschwindigkeit v_{uz} zu. In Abbildung 4.3 wird durch die Admittanz

$$A_z(s) = \frac{v_{uz}(s)}{e_{F_z}(s)}, \quad \text{mit } A_z(s) = \frac{1}{K} \cdot K_P \cdot \left(1 + \frac{1}{T_N \cdot s} + T_D \cdot s \right) \quad (4.1)$$

dem Regelfehler e_{F_z} eine gewünschte Endeffektorgeschwindigkeit v_{uz} zugeordnet. Somit eignet sich die Regelung dazu, Stellwerte \mathbf{x}_u für die Schnittstelle zur kartesischen Positionsvorgabe des Roboters zu generieren¹⁾. Die Admittanz \mathbf{A} bezieht die Steifigkeit K der Umgebung mit ein, sodass bei sehr steifer Umgebung (großes K) niedrige Geschwindigkeiten \mathbf{v}_u erzeugt werden, und bei sehr weicher Umgebung (kleines K) höhere Geschwindigkeiten erreicht werden. Die Steifigkeit K der Umgebung wird experimentell ermittelt und ergibt in Versuchen eine Steifigkeit von z.B. $K = 1278 \frac{\text{N}}{\text{m}}$.

¹⁾Für eine *kraftbasierte explizite Regelung* wäre die direkte Vorgabe der Gelenkdrehmomente notwendig. Die Steuerung des KUKA KR60 erlaubt diesen Zugriff nicht.

Einfache Einstellregeln zur empirischen Auslegung der Reglerparameter existieren z.B. von Ziegler und Nichols [LW07]. Für diese Einstellregeln wird der Regler durch ein Proportionalglied mit dem Verstärkungsfaktor K_P ersetzt. Dann wird K_P so lange erhöht, bis der Regelkreis bei $K_P = K_{P,krit}$ die kritische Stabilitätsgrenze erreicht und sich eine Dauerschwingung mit der Periodendauer T_{krit} einstellt. Aus $K_{P,krit}$ und T_{krit} werden über Faustformeln die Reglerparameter bestimmt. Da der Betrieb an der Stabilitätsgrenze zu unkontrollierbaren Schwingungen führen kann, modifizieren Åström und Hägglund [AH84] das Verfahren und verwenden einen Zweipunktreger statt des Proportionalgliedes. Dadurch stellt sich eine kontrollierbare Dauerschwingung ein, aus welcher $K_{P,krit}$ und T_{krit} ermittelt werden können. Dieses Faustformelverfahren ist nicht unbedingt geeignet für eine schnelle Kraftregelung – der Fokus dieser Arbeit liegt zum einen nicht auf der Kraftregelung, zum anderen werden bei den Experimenten zur Konturverfolgung mit Kontakt relativ niedrige Geschwindigkeiten in Kontakttrichtung gefordert, sodass diese einfache Auslegung ausreicht.

Die Kraftmessung \mathbf{F}_m des KMS beinhaltet statische Kräfte durch Gravitation \mathbf{F}_g und dynamische Kräfte (Trägheitskräfte) \mathbf{F}_{dyn} . Diese Anteile zählen nicht zur Kontaktkraft und können z.B. bei Beschleunigung den Regelkreis destabilisieren. Um ausschließlich die Kontaktkraft \mathbf{F} zu ermitteln, wird im folgenden Abschnitt die Kraftkompensation vorgestellt.

4.2 Ermittlung der Kontaktkraft

Sind Kraftregelung und bildgestützte Regelung gleichzeitig aktiv, dann besteht eine Verkopplung zwischen den Regelkreisen, da die Kommandierung einer bildgestützten Bewegung die Kraftmessung durch Trägheitskräfte beeinflusst. Wird von einem idealen Roboter ausgegangen und das Regelungsschema *Hybrid-* bzw. *Traded Control* eingesetzt, dann entsteht keine Verkopplung, da in einer Achsrichtung jeweils nur Kraftregelung **oder** bildgestützte Regelung aktiv sind [NMK96]. In den Experimenten des folgenden Abschnittes wird jedoch deutlich, dass nicht von einem idealen Roboter ausgegangen werden kann, und somit die Entkopplung durch Kraftkompensation unabhängig von der Regelungsstruktur durchgeführt werden muss.

Der KMS liefert Messwerte der Kräfte $\mathbf{F}_m = (F_{mx}, F_{my}, F_{mz})^T$ und Momente $\mathbf{M}_m = (M_{mx}, M_{my}, M_{mz})^T$. Die statischen Anteile durch Gravitati-

on \mathbf{F}_g und \mathbf{M}_g sowie die dynamischen Anteile aus Trägheits- und Zentrifugalkraft \mathbf{F}_{dyn} und \mathbf{M}_{dyn} müssen berechnet und zur Kompensation von der Kraftmessung subtrahiert werden. Daraus ergeben sich die Kontaktkräfte bzw. Kontaktmomente

$$\mathbf{F} = \mathbf{F}_m - \mathbf{F}_g - \mathbf{F}_{dyn} \quad (4.2)$$

$$\mathbf{M} = \mathbf{M}_m - \mathbf{M}_g - \mathbf{M}_{dyn} \quad . \quad (4.3)$$

Die Berechnung der statischen Anteile \mathbf{F}_g und \mathbf{M}_g sowie der dynamischen Anteile \mathbf{F}_{dyn} und \mathbf{M}_{dyn} wird in den folgenden zwei Abschnitten gezeigt.

4.2.1 Gravitationskräfte und -momente

Aufgrund der Gravitation und des Werkzeuggewichts m_t werden auch im stationären Zustand Kräfte und Momente gemessen, die aus keiner Kontaktkraft resultieren (siehe Abbildung 4.1). Die Einflüsse durch Gravitation sind von der Orientierung des Werkzeugkoordinatensystems im Verhältnis zum Basiskoordinatensystem abhängig. Die Robotersteuerung stellt diese Orientierung mittels der Rotationen $R_z(\alpha)$, $R_y(\beta)$ und $R_x(\gamma)$ um die z -, y - und x -Achse in der Konvention der *Roll-Pitch-Yaw-Winkel* (RPY) zur Verfügung. Diese Winkel werden zur Berechnung einer Rotationsmatrix

$$\mathbf{R}_{base}^{TCP} = R_z(\alpha)R_y(\beta)R_x(\gamma) = \begin{pmatrix} c_\alpha c_\beta & c_\alpha s_\beta s_\gamma - s_\alpha c_\gamma & c_\alpha s_\beta c_\gamma + s_\alpha s_\gamma \\ s_\alpha c_\beta & s_\alpha s_\beta s_\gamma + c_\alpha c_\gamma & s_\alpha s_\beta c_\gamma - c_\alpha s_\gamma \\ -s_\beta & c_\beta s_\gamma & c_\beta c_\gamma \end{pmatrix}$$

mit $c_\gamma = \cos(\gamma)$, $s_\beta = \sin(\beta)$, usw.

(4.4)

zwischen Basiskoordinatensystem und Werkzeugkoordinatensystem verwendet [SSVO09]. Die z -Achse des Basiskoordinatensystems ist in den Experimenten so ausgerichtet, dass die Gravitation g in Richtung dieser Achse wirkt. Daraus ergibt sich die Gravitationskraft und das Gravitationsmoment

$$\mathbf{F}_g = (\mathbf{R}_{base}^{TCP})^T \cdot \begin{pmatrix} 0 \\ 0 \\ m_t \cdot g \end{pmatrix} = \begin{pmatrix} -s_\beta \\ c_\beta s_\gamma \\ c_\beta c_\gamma \end{pmatrix} \cdot m_t \cdot g \quad (4.5)$$

$$\mathbf{M}_g = \mathbf{c} \times \mathbf{F}_g \quad , \quad (4.6)$$

mit $\mathbf{c} = (c_x, c_y, c_z)^T$ dem Masseschwerpunkt im Sensorkoordinatensystem.

Tabelle 4.1: Parameter zur dynamischen Kraftkompensation [WS07].

| | |
|--------------------------------|---|
| Translatorische Beschleunigung | $\mathbf{a} = (a_x, a_y, a_z)^T$ |
| Rotatorische Beschleunigung | $\boldsymbol{\alpha} = (\alpha_x, \alpha_y, \alpha_z)^T$ |
| Rotatorische Geschwindigkeit | $\boldsymbol{\omega} = (\omega_x, \omega_y, \omega_z)^T$ |
| Trägheitstensor | $\mathbf{J} = \begin{pmatrix} J_{xx} & J_{xy} & J_{xz} \\ J_{yx} & J_{yy} & J_{yz} \\ J_{zx} & J_{zy} & J_{zz} \end{pmatrix}$ |
| Massenschwerpunkt | $\mathbf{c} = (c_x, c_y, c_z)^T$ |
| Werkzeugmasse | m_t |

4.2.2 Dynamische Kräfte und Momente

Während der Bewegung und der Beschleunigung des Systems werden dynamische Kraftanteile gemessen, die aus keiner Kontaktkraft resultieren. Diese Kräfte setzen sich aus Trägheitskräften durch lineare und rotatorische Beschleunigung sowie der Zentrifugal- und Corioliskraft zusammen. Um diese Anteile zu ermitteln, sind Messwerte und Parameter nach Tabelle 4.1 notwendig.

Aus den aktuellen translatorischen und rotatorischen Beschleunigungen sowie den rotatorischen Geschwindigkeiten ergeben sich entsprechend Tabelle 4.1 und 4.2 die dynamischen Kräfte und Momente

$$\mathbf{F}_{dyn} = \mathbf{F}_a + \mathbf{F}_\alpha + \mathbf{F}_c \quad (4.7)$$

$$\mathbf{M}_{dyn} = \mathbf{M}_a + \mathbf{M}_\alpha + \mathbf{M}_c \quad . \quad (4.8)$$

Die Masse des Werkzeuges, der Massenschwerpunkt und der Trägheitstensor können z.B. aus CAD Daten oder aus Messwerten ermittelt werden [WS07]. Für diese Arbeit wird die Masse und der Massenschwerpunkt messtechnisch aus Kräften und Momenten in unterschiedlichen Orientierungen des Werkzeuges ermittelt. Die Ermittlung des Trägheitstensors erfordert bei einer beschleunigten Bewegung des Roboters die Messung der Momente und der rotatorischen Beschleunigung.

Tabelle 4.2: Dynamische Anteile der Kräfte und Momente [WS07].

| Ursache | Kraft | Moment |
|--------------------------------|---|--|
| Translatorische Beschleunigung | $\mathbf{F}_a = m_t \cdot \mathbf{a}$ | $\mathbf{M}_a = \mathbf{c} \times \mathbf{F}_a$ |
| Rotatorische Beschleunigung | $\mathbf{F}_\alpha = m_t \cdot (\boldsymbol{\alpha} \times \mathbf{c})$ | $\mathbf{M}_\alpha = \mathbf{J} \cdot \boldsymbol{\alpha}$ |
| Rotatorische Geschwindigkeit | $\mathbf{F}_c = m_t \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{c})$ | $\mathbf{M}_c = \boldsymbol{\omega} \times (\mathbf{J} \cdot \boldsymbol{\omega})$ |

Im folgenden Abschnitt wird die Messung der translatorischen und rotatorischen Beschleunigung mit unterschiedlichen Sensorsystemen untersucht. Die Kompensation der dynamischen Kräfte wird im darauf folgenden Kapitel experimentell gezeigt.

4.3 Beschleunigungsmessung

Zur Kompensation der Trägheitskräfte \mathbf{F}_a und \mathbf{F}_α müssen die aktuellen translatorischen Beschleunigungen \mathbf{a} und rotatorischen Beschleunigungen $\boldsymbol{\alpha}$ ermittelt werden. Die Anbringung der Beschleunigungssensorik am Roboter wird in Abbildung 4.4 gezeigt. Es werden die drei nachfolgend beschriebenen Methoden zur Ermittlung der Beschleunigung verglichen:

- **Methode 1:** Doppelte Ableitung der Roboterpositionsdaten
- **Methode 2:** Direkte Messung am Roboterhandflansch (12D KMS)
- **Methode 3:** Direkte Messung am Werkzeug (drei 2-Achs Sensoren)

Die translatorische Beschleunigung $\mathbf{a}_{rob} = (a_{robx}, a_{robz}, a_{robz})^T$ und die rotatorische Beschleunigung $\boldsymbol{\alpha}_{rob} = (\alpha_{robx}, \alpha_{robz}, \alpha_{robz})^T$ werden nach **Methode 1** aus der kartesischen Position und Orientierung

$$\mathbf{r}_k = (r_{xk}, r_{yk}, r_{zk}, r_{\gamma k}, r_{\beta k}, r_{\alpha k})^T, \quad k = 0, 1, \dots, M \quad (4.9)$$

des Roboters berechnet. Die Roboterdaten werden aus der Vorwärtskinematik in Abhängigkeit der Gelenkwinkel des Roboters im Interpolations-

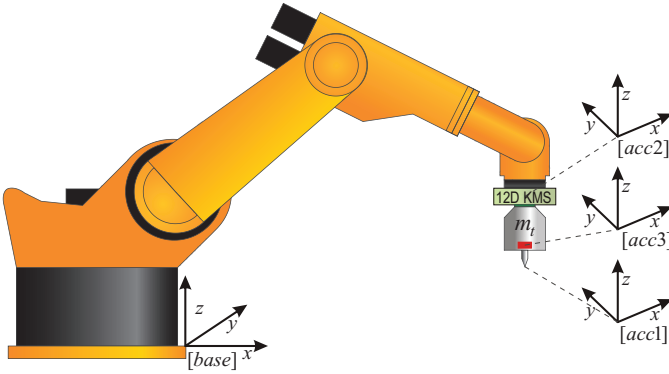


Abbildung 4.4: Beschleunigungssensorik am Roboter. Die Beschleunigung kann aus Roboterdaten ermittelt werden $[acc1]$, aus der Messung des 12D KMS $[acc2]$ oder durch Messung direkt am Werkzeug $[acc3]$.

takt $T_{IPO} = 4ms$ berechnet. Über den Differenzenquotienten ergeben sich die Beschleunigungen zu:

$$\begin{pmatrix} \mathbf{a}_{rob,k} \\ \boldsymbol{\alpha}_{rob,k} \end{pmatrix} = \frac{\mathbf{r}_k - 2 \cdot \mathbf{r}_{k-1} + \mathbf{r}_{k-2}}{T_{IPO}^2} \quad . \quad (4.10)$$

Die Messung der Beschleunigung \mathbf{a}_{KMS} bzw. $\boldsymbol{\alpha}_{KMS}$ nach **Methode 2** erfolgt mittels Beschleunigungssensoren²⁾, welche im 12D KMS verbaut sind. Das Gehäuse des 12D KMS ist am Roboterflansch befestigt; die Beschleunigungssensoren sind am Gehäuse montiert. Daraus folgt, dass die Beschleunigung des Roboterhandflansches gemessen wird.

Die Kraft-Momenten Sensorik des 12D KMS ist nachgiebig aufgebaut, sodass sich der werkzeugseitige Flansch bei Belastung um $\pm 1,4mm$ gegenüber dem Gehäuse bewegen kann. Durch diese Nachgiebigkeit entsteht eine zusätzliche Dynamik zwischen Werkzeug und Gehäuse, sodass \mathbf{a}_{KMS} bzw. $\boldsymbol{\alpha}_{KMS}$ nicht die exakten Beschleunigungen des Werkzeuges beschreiben. Um den Unterschied zwischen Methode 2 und der tatsächlichen Beschleunigung des Werkzeuges zu beurteilen, werden Beschleunigungssensoren³⁾

²⁾Intern werden drei 2-Achs Sensoren ADXL202 von Analog Devices verwendet mit jeweils einem Messbereich von $\pm 2g$.

³⁾Die Beschleunigungssensoren ADXL322J mit einem Messbereich von $\pm 2g$ werden mit 10Bit bei 500Hz Abtastfrequenz ausgelesen.

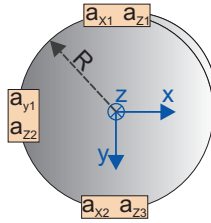


Abbildung 4.5: Anbringung der drei 2-Achs Beschleunigungssensoren am werkzeugseitigen Flansch des KMS auf einem Radius $R = 100mm$.

auf dem Werkzeug montiert (**Methode 3**). Die Anordnung der drei 2-Achs Sensoren ist in Abbildung 4.5 dargestellt. Die Sensoren sind auf einem Radius von $R = 100mm$ angebracht. Die translatorischen und rotatorischen Beschleunigungen \mathbf{a}_t bzw. $\boldsymbol{\alpha}_t$ des Werkzeuges werden daraus folgendermaßen berechnet:

$$\begin{aligned}
 \mathbf{a}_t &= (a_{tx}, a_{ty}, a_{tz})^T & \boldsymbol{\alpha}_t &= (\alpha_{tx}, \alpha_{ty}, \alpha_{tz})^T \\
 a_{tx} &= \frac{a_{x1} + a_{x2}}{2} & \alpha_{tx} &= \frac{a_{z3} - a_{z1}}{2R} \\
 a_{ty} &= a_{y1} - \frac{a_{x1} - a_{x2}}{2} & \alpha_{ty} &= \frac{a_{z2} - a_{tz}}{R} \\
 a_{tz} &= \frac{a_{z1} + a_{z3}}{2} & \alpha_{tz} &= \frac{a_{x1} - a_{x2}}{2R} .
 \end{aligned} \tag{4.11}$$

Die Messung der Beschleunigung \mathbf{a}_t bzw. \mathbf{a}_{KMS} beinhaltet auch die Messung der Gravitation g . Dieser Anteil muss vom Messwert subtrahiert werden. Analog zu Gleichung 4.5 wird die Gravitationsbeschleunigung \mathbf{a}_g abhängig von der aktuellen Orientierung des Endeffektors berechnet:

$$\begin{aligned}
 \mathbf{a}_g &= (\mathbf{R}_{base}^{TCP})^T \cdot (0 \quad 0 \quad g)^T ; & \mathbf{a}_{tg} &= \mathbf{a}_t - \mathbf{a}_g \\
 & & \mathbf{a}_{KMSg} &= \mathbf{a}_{KMS} - \mathbf{a}_g .
 \end{aligned} \tag{4.12}$$

Um in diesem Abschnitt zunächst die Sensorik zu beurteilen, wurden die werkzeugseitigen Beschleunigungssensoren (Methode 3) zunächst nicht am Werkzeug, sondern am Gehäuse des KMS befestigt. Die direkte Messung durch Methode 2 und 3 muss somit identische Werte liefern. Abbildung 4.6(a) zeigt die translatorische Beschleunigung bei sprungförmiger Vorgabe der z -Position, während der Roboter eine lineare Bewegung in

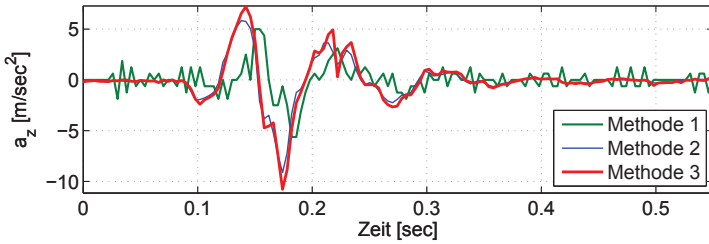
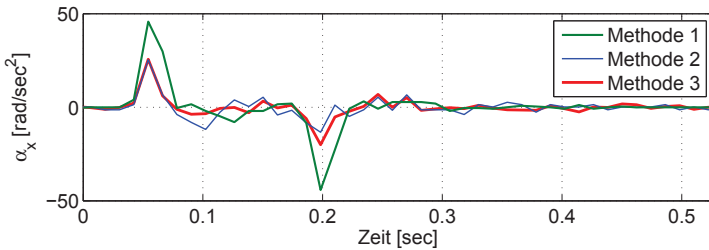
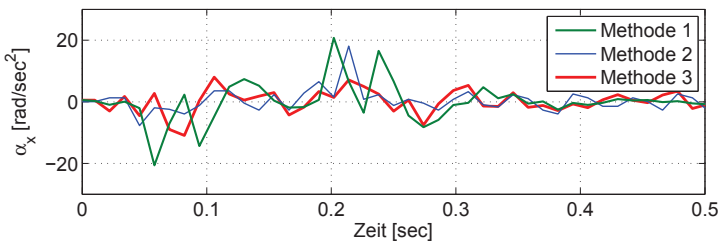
(a) Translatorische Beschleunigung in z -Richtung(b) Rotatorische Beschleunigung um x -Achse aus Stillstand(c) Rotatorische Beschleunigung um x -Achse aus Bewegung

Abbildung 4.6: Vergleich der Beschleunigung durch Messmethode 1-3 bei sprungförmiger Bewegung des Roboters. Zur Beurteilung der Sensorik wurden die Sensoren nach Methode 3 hier nicht am Werkzeug, sondern am Gehäuse des KMS montiert.

x -Richtung des Basiskoordinatensystems durchführt. Es ist deutlich zu erkennen, dass beide Beschleunigungssensoren (Methoden 2 und 3) vergleichbare Werte liefern. Die doppelte Ableitung der Roboterposition (Methode 1) liefert dagegen ein stark verrauschtes Signal aufgrund von Diskretisierungsrauschen und der doppelten Ableitung von Störeinflüssen.

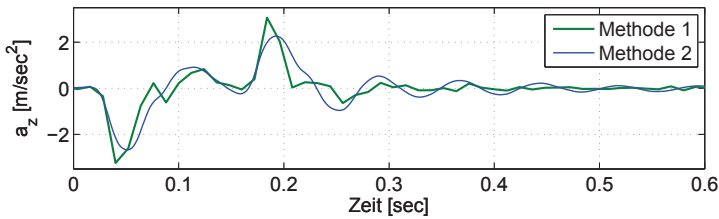
In Abbildung 4.6(b) wurde eine sprungförmige Änderung der Orientierung um die x -Achse vorgegeben. Die Messung zeigt die Beschleunigung deutlich, und die direkten Messmethoden (Methode 2 und 3) liefern vergleichbare Ergebnisse.

Beschleunigungssensoren sind relativ empfindlich gegenüber Vibrationen. Die Rotationsbeschleunigung wird aus der Überlagerung von bis zu drei Sensorsignalen berechnet (vgl. Gleichung 4.11). Verfolgt der Roboter mit konstanter Geschwindigkeit eine beliebige Trajektorie im Raum, sind meistens alle sechs Motoren in Bewegung und erzeugen Vibrationen. Wird die sprungförmige Orientierungsänderung während einer linearen Bewegung des Roboters durchgeführt, ist die Messung der Rotationsbeschleunigung extrem verrauscht, und die direkten Messmethoden liefern auch keine vergleichbaren Ergebnisse mehr (siehe Abbildung 4.6(c)).

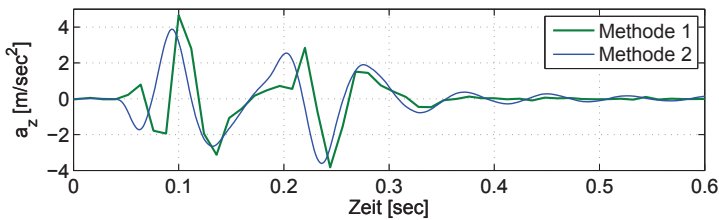
Für die nachfolgenden Experimente zur Kompensation von dynamischen Kräften wird die Beschleunigungssensorik wieder wie in Abbildung 4.4 dargestellt am Werkzeug montiert. Aufgrund des starken Sensorrauschens der rotatorischen Beschleunigungssignale beschränken sich die Experimente auf die Kompensation der Trägheitskräfte \mathbf{F}_a durch translatorische Beschleunigung. Zudem wird bei der Konturverfolgung aufgrund der begrenzten rotatorischen Geschwindigkeiten erwartet, dass der Einfluss der rotatorischen Beschleunigung \mathbf{F}_α und der Zentrifugalkraft \mathbf{F}_c im Vergleich zu \mathbf{F}_a einen nur geringen Anteil an der dynamischen Kraft \mathbf{F}_{dyn} haben.

4.4 Kompensation von Trägheitskräften

Drei Methoden zur Ermittlung translatorischer Beschleunigung wurden im vorangegangenen Abschnitt vorgestellt. In diesem Abschnitt wird zunächst die aus den Roboterdaten berechnete Beschleunigung (Methode 1) mit der direkten Messung der Beschleunigung am Handflansch des Roboters (Methode 2) verglichen. Im Anschluss wird der Einfluss der Positionierung der Beschleunigungssensoren (Methode 3) untersucht.



(a) Messung **parallel** zur Bewegungsrichtung: die Bewegung wird in z -Richtung kommandiert und die Beschleunigung in z -Richtung gemessen.



(b) Messung **orthogonal** zur Bewegungsrichtung: die Bewegung wird in x -Richtung kommandiert und die Beschleunigung in z -Richtung gemessen.

Abbildung 4.7: Vergleich der Beschleunigungssignale zwischen Methode 1 und 2. Zwischen den Signalen besteht eine Zeitverzögerung (b), welche von der Bewegungsrichtung abhängt.

4.4.1 Berechnung der Trägheitskraft aus Roboterdaten

Die Messung der translatorischen Beschleunigung in z -Richtung nach Methode 1 und 2 wird in Abbildung 4.7 verglichen. Es werden zwei Fälle untersucht:

- (a) Die sprungförmige Bewegung des Roboters wird in z -Richtung kommandiert und die Beschleunigung in z -Richtung gemessen. Die Messung erfolgt in der Richtung der kommandierten Bewegung.
- (b) Die sprungförmige Bewegung des Roboters wird in x -Richtung kommandiert und die Beschleunigung in z -Richtung gemessen. Die Messung erfolgt orthogonal zur kommandierten Bewegungsrichtung.

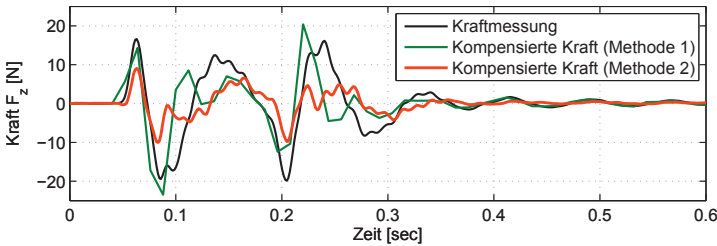
Beide Methoden liefern für beide Fälle ungefähr vergleichbare Signalverläufe, jedoch mit unterschiedlicher Signalverzögerung. Im Fall (a) eilt die Messung nach Methode 1 leicht voraus, im Fall (b) eilt die Messung nach Methode 2 voraus. Würde von einem idealen Roboter ausgegangen, dann dürfte im Fall (b) keine Beschleunigung gemessen werden, da die Messung orthogonal zur Bewegung durchgeführt wird. Gerade bei hohen Beschleunigungen treten jedoch erhebliche Vibrationen in allen Raumrichtungen auf.

Die unterschiedlichen Signalverzögerungen resultieren aus der Dynamik zwischen Antrieb und Abtrieb, welche durch die endliche Steifigkeit des Antriebsstranges hervorgerufen werden⁴⁾:

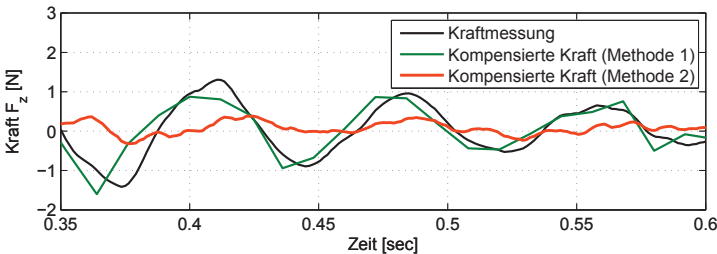
- (a) Da die Beschleunigung in der gewünschten Bewegungsrichtung gemessen wird, wird die entsprechende Bewegung der Antriebsmotoren gemessen, bevor die tatsächliche Bewegung durch den Antriebsstrang den Endeffektor erreicht.
- (b) Die Beschleunigung wird quer zur Richtung der gewünschten Bewegung gemessen, sodass lediglich die in der kinematischen Kette auftretenden Vibrationen gemessen werden. Diese Vibrationen müssen durch den Antriebsstrang laufen, bevor sie durch die Messung der Gelenkwinkel am Antriebsmotor erfasst werden können. In diesem Fall ist die Beschleunigungsmessung nach Methode 1 um ca. $25ms$ gegenüber der Messung nach Methode 2 verschoben.

Die Kompensation der Trägheitskräfte nach Methode 1 und 2 ist vergleichend in Abbildung 4.8 dargestellt. Für dieses Experiment wurde eine sprungförmige Bewegung des Roboters in x -Richtung kommandiert, während die Kräfte in z -Richtung aufgezeichnet wurden. Da es sich bei dem Experiment um eine Bewegung ohne Kontakt handelt, müsste eine ideale Kompensation einen Kraftverlauf bei konstant $0N$ erzeugen. Da es sich hier um eine extrem ruckartige Bewegung handelt, gelingt die vollständige Kompensation nicht. Durch die Signalverzögerung des Beschleunigungssignales verschlechtert die Kompensation nach Methode 1 das Ergebnis sogar. Die Kompensation nach Methode 2 ist deutlich effektiver. In Abbildung 4.8(b) wird das Nachschwingen des Roboters nach dieser ruckartigen Bewegung durch Methode 2 fast vollständig kompensiert, wohingegen die Kompensation auf Basis der Roboterdaten (Methode 1) diese Restschwingung kaum erfassen kann, und somit das Signal nahezu unverändert bleibt.

⁴⁾Der prinzipielle Aufbau eines Antriebsstranges wurde in Abbildung 2.2 (Seite 44) vorgestellt.



(a) Die Kompensation der Trägheitskraft nach Methode 2 ist bei hoher Beschleunigung besser als nach Methode 1.

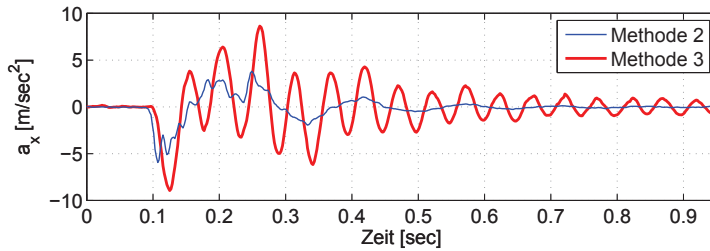


(b) Ausschnitt aus Abbildung (a). Die „Restschwingungen“ werden durch Methode 2 fast vollständig kompensiert.

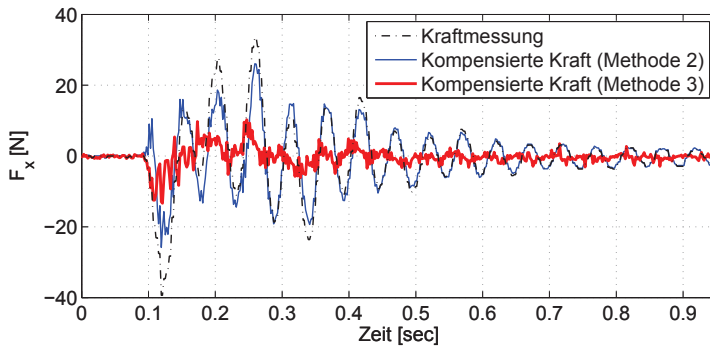
Abbildung 4.8: Vergleich der Kraftkompensation nach Methode 1 und 2. Eine sprungförmige Bewegung des Roboters wurde in x -Richtung kommandiert, die Kräfte werden in z -Richtung gemessen.

4.4.2 Berechnung der Trägheitskraft aus Beschleunigungssensoren

Der direkte Sensorvergleich in Abbildung 4.6 hat gezeigt, dass die Sensoren zur direkten Beschleunigungsmessung (Methode 2 und 3) vergleichbare Ergebnisse produzieren, wenn sie an der gleichen Stelle montiert sind. Wenn der nachgiebige KMS verwendet wird, dann kann das Werkzeug aufgrund der Nachgiebigkeit nach einer Beschleunigungsphase Schwingungen ausführen, welche als Trägheitskräfte gemessen werden. Die Kompensation dieser Kräfte aufgrund einer Beschleunigungsmessung am Roboterflansch (Methode 2) ist unmöglich, da diese Bewegung dort nicht messbar ist. Da-



(a) Beschleunigungssignal



(b) Kraftkompensation

Abbildung 4.9: Vergleich der Beschleunigungsmessung von Methode 2 und 3. Die Kompensation nach Methode 3 liefert ein besseres Ergebnis bei Schwingungen am nachgiebigen KMS nach ruckartiger Bewegung.

her wurden die Beschleunigungssensoren nach Methode 3 direkt auf dem Werkzeug montiert, um dessen Beschleunigung zu erfassen.

Die Beschleunigungssignale bei abruptem Stoppen einer linearen Bewegung des Roboters sind in Abbildung 4.9(a) dargestellt. Es ist deutlich, dass die Sensoren auf dem Werkzeug mit der Masse $m_t = 5\text{ kg}$ eine deutliche Schwingung erfassen, wohingegen die Sensoren nach Methode 2 lediglich die ruckartige Bewegung des Roboterflansches erfassen können. Die zugehörige Kraftmessung ist in Abbildung 4.9(b) dargestellt. Die Kompensation nach Methode 2 hat nahezu keinen Effekt, wohingegen die Kompensation nach Methode 3 ein deutlich stärker kompensiertes Signal erzeugt.

4.5 Zusammenfassung

In diesem Kapitel wurde die Kraftregelung von Robotern vorgestellt. Es wurde die positionsbasierte explizite Kraftregelung (Admittanzregelung) vorgestellt. Diese Art der Regelung wurde ausgewählt, da das Robotersystem nur über eine Schnittstelle zur kartesischen Positionsvorgabe verfügt. Als Regler wurde ein PID-Regler eingesetzt.

Der Fokus dieses Kapitels lag nicht bei der Kraftregelung selbst, sondern bei der Kompensation von Gravitations- und Trägheitskräften in der Kraftmessung des KMS. Die Gravitationskraft wurde mithilfe der aus der Robotersteuerung verfügbaren Orientierung des Endeffektorkoordinatensystems berechnet und von den Kraftmesswerten subtrahiert. In Beschleunigungsphasen des Roboters treten Trägheitskräfte auf, welche nicht zur Kontaktkraft zählen. Für die Ermittlung dieser Kräfte ist die Kenntnis der aktuellen Beschleunigung notwendig, welche auf drei verschiedene Methoden ermittelt wurde.

Die erste Methode basiert auf der Berechnung der zweiten Ableitung der Roboterpositionsdaten. Dieser Messwert ist zum einen stark durch Quantisierungsrauschen gestört, und zum anderen wird durch Nachgiebigkeiten im Antriebsstrang eine nicht-konstante Verzögerung zwischen der tatsächlichen und der berechneten Beschleunigung festgestellt.

Methoden 2 und 3 basieren auf der direkten Messung der Beschleunigung mittels Beschleunigungssensoren. Bei Verwendung eines nachgiebigen KMS haben sich Dynamiken gezeigt, welche am besten durch Methode 3 berücksichtigt werden können, da diese im Vergleich zu Methode 2 die Beschleunigung nicht am Roboterflansch, sondern am Werkzeug selber ermittelt. Die Rotationsbeschleunigung bei Methode 2 und 3 wird aus der Messung der translatorischen Beschleunigung berechnet. Aufgrund der Empfindlichkeit der Sensoren auf Vibrationen ist das Signal so stark verrauscht, dass es nicht zu Kompensation verwendet werden konnte. Somit beschränkten sich die Experimente in diesem Kapitel ausschließlich auf die eingehende Untersuchung der Kompensation von Trägheitskräften durch translatorische Beschleunigung.

Insgesamt wurde deutlich, dass die Positionierung der Beschleunigungssensoren einen gravierenden Einfluss auf das Ergebnis der Kompensation hat. Je näher der Sensor am Werkzeug platziert wird (Methode 3), desto genauer wird die Kompensation.

5 Konturverfolgung durch Fusion von Kraft- und Bildsensorik

In den vorangegangenen Kapiteln wurden die bildgestützte Regelung und die Kraftregelung sowie die Messwerterfassung vorgestellt. Dabei wurden die Regelkreise ausschließlich unabhängig voneinander betrachtet. Um die beiden Regelkreise gleichzeitig einzusetzen, müssen einige Vorüberlegungen hinsichtlich Sensorsynchronisation und Entkopplung getroffen werden, da die Regelung Informationen aus unterschiedlichen Signalquellen miteinander vereint. Zu den Signalquellen zählen:

- Kartesische Roboterposition \mathbf{r} .
- Bildmerkmal \mathbf{p} aus dem Laser-Triangulationssensor.
- Kraftmessung \mathbf{F}_m des KMS.
- Beschleunigung \mathbf{a} aus Beschleunigungssensorik.

In Multisensorapplikationen spielt die Synchronisation von Sensorsystemen eine große Rolle. Olson [Ols10] beschreibt die Problematik der Sensorsynchronisation damit, dass den meisten Sensoren die Möglichkeit fehler, aktiv synchronisierbar zu sein. Die meisten Sensoren laufen in einem eigenen Sensortakt und können nur durch „passive Synchronisation“ aneinander angepasst werden. D.h., Zeitverzögerungen können nur dadurch erkannt werden, indem in den unterschiedlichen Signalströmen bestimmte messbare Ereignisse zur Synchronisation verwendet werden. Einige Sensoren, wie z.B. Kameras, besitzen jedoch die Fähigkeit zur aktiven Synchronisation. De Graaf et al. [GAMJ05] erreichen die Synchronisation, indem die Bildaufnahme einer Kamera im Interpolationstakt des Roboters getriggert wird. Mit bekannten Übertragungsverzögerungen wird das Bildsignal genau mit dem Robotersignal synchronisiert.

In diesem Kapitel werden neben der Sensorsynchronisation auch die Kopplungen zwischen den Regelkreisen bzw. Sensorsystemen betrachtet. Abbildung 5.1 zeigt einen Überblick über dieses Kapitel. Teile dieses Kapitels wurden in [KKK⁺11a, KKK⁺11b] veröffentlicht. Es treten prinzipiell

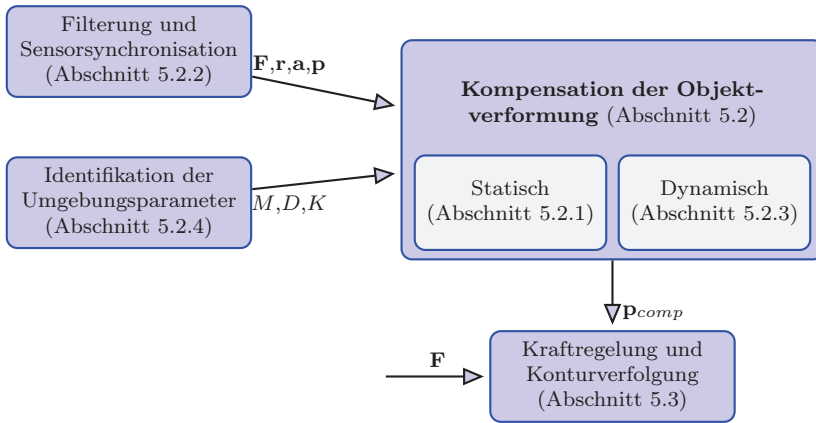


Abbildung 5.1: Übersicht mit Kapitelangabe. Die Kompensation der Objektverformung basiert auf den Umgebungsparametern Steifigkeit K , Masse M und Dämpfung D . Die Verformung wird aus der Roboterposition \mathbf{r} und der Kontaktkraft \mathbf{F} berechnet. Zur Regelung werden die kompenzierte Bildinformationen \mathbf{p}_{comp} und die Kraftmessung verarbeitet.

zwei Arten von Verkopplungen zwischen bildgestützter Konturverfolgung und Kraftregelung auf:

1. Beschleunigungen, welche bildgestützt kommandiert werden, resultieren in **Trägheitskräften**, welche der KMS misst.
2. **Verformungen**, welche durch eine sich ändernde Kontaktkraft an einem nachgiebigen Objekt auftreten, verändern die Position von sichtbaren Merkmalen auf der Kontur.

Die Kompensation der Trägheitskräfte zur Ermittlung der Kontaktkraft wurde in Kapitel 4 betrachtet. Somit kann der Kraftregelkreis unabhängig von Bewegungen des Roboters und damit unabhängig von Bewegungen aus bildgestützter Regelung betrieben werden.

Auf Verkopplungen durch Verformungen bei Kontakt an nachgiebigen Werkstücken wird in diesem Kapitel eingegangen. Die Verkopplungen entstehen dadurch, dass das Kamerasystem die Kontur vorausschauend erfasst. Somit führt eine Verformung des Materials bei der Konturverfolgung dazu, dass die im Vorlauf aufgezeichneten Daten nicht dem tatsächlichen Verlauf der Kontur nach Verformung entsprechen. Die Kompensation die-

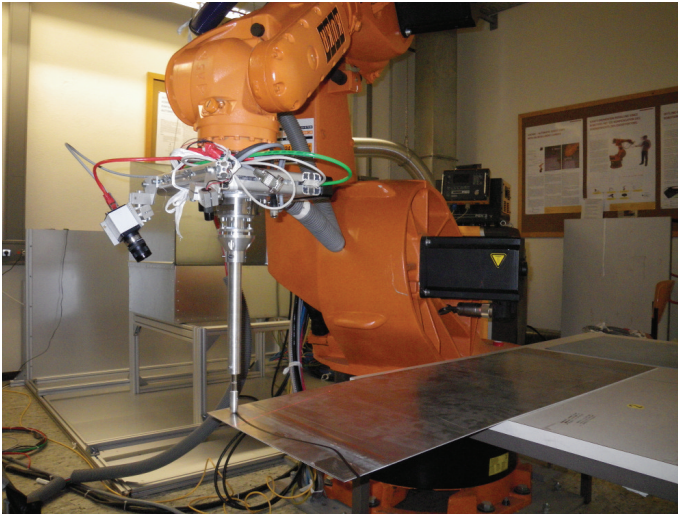


Abbildung 5.2: Experimentalsystem mit einer sichtbaren Kontur auf einem einseitig eingespannten Metallblech zur Konturverfolgung mit Objektkontakt an nachgiebigen Werkstücken.

ser Verformung durch Modellierung des Werkstückes bei Kontakt wird vorgestellt und die dazu notwendigen Parameter werden durch online Identifikation ermittelt. Danach werden die Ergebnisse an unterschiedlichen Konturen validiert. Abbildung 5.2 zeigt den Systemaufbau mit einer Kontur auf einem nachgiebigen Metallblech. Die dargestellte Kontur soll bei wechselnder Kontaktkraft mit konstanter Orientierung des Werkzeuges zur Kontur verfolgt werden.

5.1 Einfluss der Objektverformung auf die Konturverfolgung mit Kontakt

Zur Beschreibung der Verformung von Werkstücken durch Kontaktkraft wird zwischen zwei Verformungstypen unterschieden, welche schematisch in Abbildung 5.3 dargestellt sind:

- **Globale Objektverformung:** Durch hohe Steifigkeit und die Art der Einspannung verformt sich das Werkstück bzw. die Kontur durch Kon-

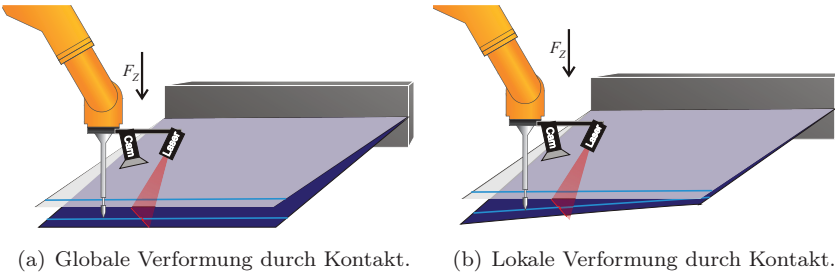
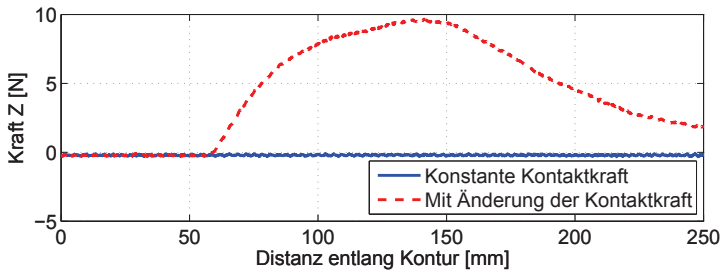


Abbildung 5.3: Abhängig von Befestigung und Steifigkeit verformt sich das Werkstück im Sichtfeld der Kamera gleichmäßig (a) oder ungleichmäßig (b). Bei ungleichmäßiger Verformung unterscheidet sich die Auslenkung im Bildpunkt der Kamera von der Auslenkung im Kontaktpunkt.

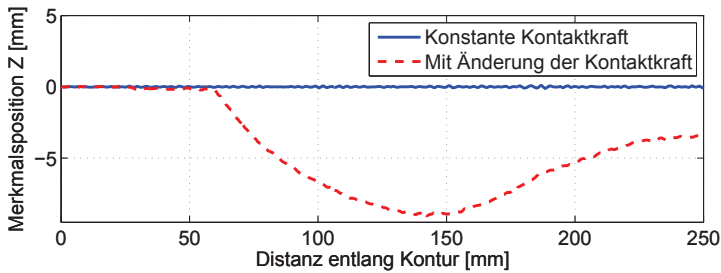
takt gleichmäßig innerhalb des Sichtfeldes der Kamera. D.h., die Auslenkung im Kontaktpunkt ist mit der Auslenkung im vorausschauenden Messpunkt der Kamera identisch (Abbildung 5.3(a)).

- **Lokale Verformung:** Durch geringe Steifigkeit bildet sich (meist zusätzlich zu einer globalen Verformung) eine Verbiegung des Materials um die Kontaktstelle herum aus. D.h., die Auslenkung im Kontaktpunkt ist ungleich der Auslenkung im vorausschauenden Messpunkt der Kamera (Abbildung 5.3(b)).

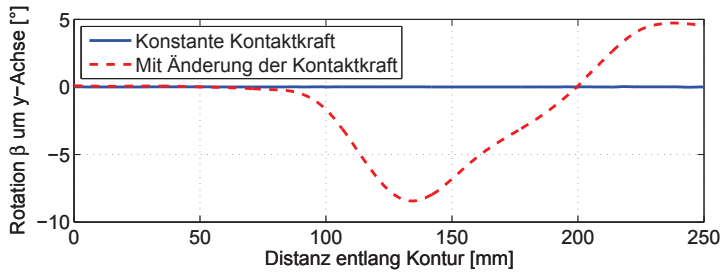
Mit dem Laser-Triangulationssensor werden Merkmale der zu verfolgenden Kontur im Vorlauf ermittelt (vgl. Kapitel 3). Abbildung 5.3 ist zu entnehmen, dass durch den Kontakt zwischen Werkzeug und nachgiebigem Werkstück die Position der Merkmale im Raum verändert wird. Die Veränderung der Merkmale einer geradlinigen Kontur auf einem einseitig eingespannten Metallblech ist in Abbildung 5.4 dargestellt. Die Messung erfolgte einmal mit konstanter Kontaktkraft (blau) und einmal mit veränderter Kontaktkraft (rot). Aufgrund der sich ändernden Auslenkung durch Kontakt wird in jedem Zeitschritt i die Messung \mathbf{p}_i der Kamera verschoben. Durch die sequenzielle Speicherung der Messwerte während der Bewegung entlang der geradlinigen Kontur entsteht ein scheinbar verformter Merkmalspfad \mathbf{P} (Abbildung 5.4(b) rot). Bei Anwendung des in Kapitel 3 vorgestellten Algorithmus zur Konturverfolgung an dem durch Kontakt verformten Merkmalspfad \mathbf{P} ergibt sich für die Rotation um die y -Achse (zur senkrechten Ausrichtung des Werkzeuges an die Kontur) die



(a) Kontaktkraft bei Konturverfolgung.

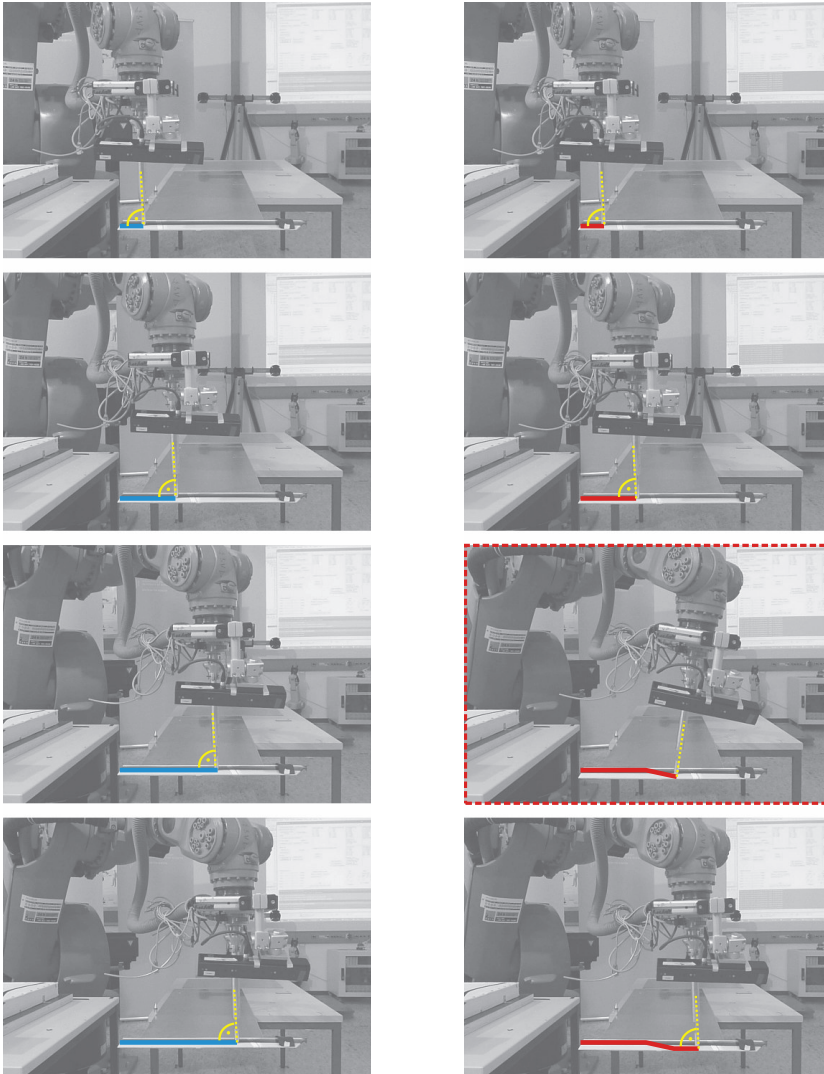


(b) Aufgezeichnete Kontur. Die Kontur erscheint stark verformt bei veränderter Kontaktkraft (rot).



(c) Orientierung des Werkzeuges. Durch die Verformung der Kontur während der Verfolgung entsteht eine Fehlstellung des Werkzeuges zur eigentlich geraden Kontur (rot).

Abbildung 5.4: Einfluss der Kontaktkraft auf die Konturverfolgung an einem einseitig eingespannten Metallblech. Die Änderung der Kontaktkraft (a) während der Verfolgung einer geradlinigen Kontur erzeugt eine hauptsächlich globale Verformung. Entlang der verformten Kontur (b) ergeben sich Fehlstellungen des Werkzeuges zur Kontur (c).



(a) Konstante Kontaktkraft.

(b) Mit Änderung der Kontaktkraft.

Abbildung 5.5: Vergleich der Werkzeugorientierung entlang der nachgiebigen Kontur. Durch die veränderte Kontaktkraft (roter Kasten) neigt sich das Werkzeug signifikant, sodass es nicht mehr senkrecht zur Kontur steht.

in Abbildung 5.4(c) dargestellte Orientierung. Die Rotationen um die y -Achse bei konstanter Kontaktkraft (blau) und wechselnder Kontaktkraft (rot) werden verglichen. Da das Werkstück hauptsächlich global verformt wird, sollte sich auch nach Änderung der Kontaktkraft keine signifikante Winkeländerung entlang der geradlinigen Kontur ergeben. Durch die Anpassung an die verformte Kontur (rot) ergeben sich jedoch Änderungen der Orientierung. Dies zeigt, dass die bildgestützte Konturverfolgung und die Kraftregelung stark über die Verformung des Werkstückes verkoppelt sind. In Abbildung 5.5 ist die Werkzeugorientierung bei konstanter und veränderter Kontaktkraft anhand einer Fotosequenz am Versuchsaufbau dargestellt. Eine Entkopplung kann durch die Kompensation der Verformung in den Bilddaten erfolgen und wird im folgenden Abschnitt betrachtet¹⁾.

5.2 Kompensation der Objektverformung

Zur Anpassung der Werkzeugausrichtung gegenüber der Kontur auf dem Werkstück werden Richtungen entlang des Merkmalspfades \mathbf{P} berechnet. Um Fehlstellungen durch Verformung bei Objektkontakt zu minimieren, wird in diesem Abschnitt der unverformte Pfad über ein Modell der Umgebung unter der aktuellen Kontaktkraft berechnet. Bei der Kompensation wird, wie schon zuvor bei der Kraftkompensation in Kapitel 4, zwischen *statischer* und *dynamischer* Kompensation unterschieden. Bei statischer Kompensation wird die Verformung mittels der Modellierung als lineare Feder unter der aktuellen Kontaktkraft berechnet. Bei dynamischer Kompensation werden zusätzlich die Trägheit und die Dämpfung des Materials mit einbezogen.

5.2.1 Statische Kompensation der Kontaktkraft

Durch die Kontaktkraft F_z in z -Richtung des Werkzeugkoordinatensystems wird eine Feder um die Strecke Δs_z ausgelenkt. Mit der bekannt-

¹⁾Die Verkopplung würde bei direktem Visual Servoing (im Gegensatz zum verwendeten Dynamic Look-And-Move) deutlich geringer ausfallen, da nicht der Verlauf der Merkmale im Raum, sondern einzelne Bildmerkmale direkt zur Regelung verwendet werden würden. D.h., bei einer globalen Objektverformung würde bei direktem Visual Servoing keine messbare Veränderung des Merkmales entstehen, da die mitgeführte Kamera und das Werkstück die gleiche Verschiebung erfahren.

ten Federsteifigkeit K kann die Auslenkung in Abhängigkeit der aktuellen Kraft F_z berechnet werden (siehe auch [BDS01]):

$$\Delta s_z = \frac{\Delta F_z}{K} \quad . \quad (5.1)$$

Im Folgenden wird für verbesserte Lesbarkeit immer davon ausgegangen, dass der Auslenkung der Feder $\Delta s_z = 0mm$ die Kontaktkraft $F_z = 0N$ zugeordnet wird. Daher wird ΔF_z durch F_z ersetzt:

$$\Delta s_z = \frac{F_z}{K} \quad . \quad (5.2)$$

Durch die Bildverarbeitung wird in jedem Zeitschritt i im Interpolationsstakt ein Messwert \mathbf{p}_i entlang der Kontur ermittelt und zum Merkmalspfad \mathbf{P} hinzugefügt. Aus Gleichung 5.2 wird die Verformung Δs_z des Werkstückes durch den Objektkontakt im Werkzeugkoordinatensystem berechnet und von den Bildmerkmalen \mathbf{p}_i subtrahiert, um die unverformte Kontur zu berechnen. Da die Bildverarbeitung Ergebnisse in Basiskoordinaten erzeugt, die Verformung jedoch in Werkzeugkoordinaten berechnet wird, muss die aktuelle Verformung in das Basiskoordinatensystem des Roboters mit der Transformation $\mathbf{T}_{base}^{TCP,i}$ umgerechnet werden:

$$\Delta \mathbf{p}_i^{TCP,i} = \begin{pmatrix} 0 & 0 & \frac{F_{z,i}}{K} \end{pmatrix}^T \quad (5.3)$$

$$\Delta \mathbf{p}_i = \mathbf{T}_{base}^{TCP,i} \cdot \Delta \mathbf{p}_i^{TCP,i} \quad (5.4)$$

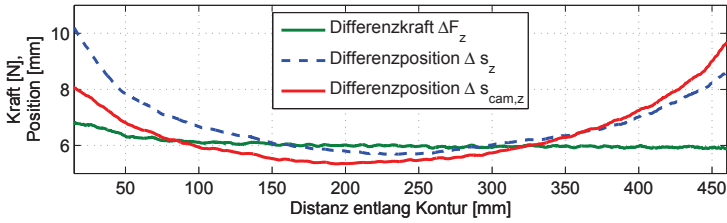
$$\mathbf{p}_{comp,i} = \mathbf{p}_i - \Delta \mathbf{p}_i \quad , \quad (5.5)$$

wobei $\mathbf{p}_{comp,i}$ die Position der Bildmerkmale nach Kompensation der Verformung beschreibt; bei idealer Kompensation entspricht dies den unverformten Merkmalen ohne Kontakt.

Da die Federsteifigkeit nicht konstant über das gesamte Werkstück ist, wird sie in Abhängigkeit der Strecke l_i entlang der Kontur beschrieben. Dadurch wird die statische Kompensation nach Gleichung 5.3 folgendermaßen erweitert:

$$\Delta \mathbf{p}_i^{TCP,i} = \begin{pmatrix} 0 & 0 & \frac{F_{z,i}}{K(l_i)} \end{pmatrix}^T \quad . \quad (5.6)$$

Die Federsteifigkeit $K(l_i)$ wird in einem offline Verfahren ermittelt, indem das Werkzeug einmal mit der Kontaktkraft $F_{1,z} \approx 2N$ und einmal mit der



(a) Differenzposition und Differenzkraft zur Ermittlung der Steifigkeit.

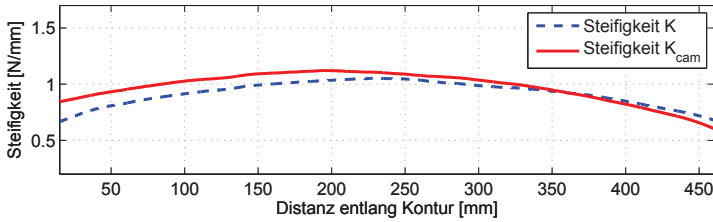
(b) Federsteifigkeit K im Kontaktpunkt und K_{cam} im Bildpunkt.

Abbildung 5.6: Berechnung der Federsteifigkeit am Metallblech aus der Differenzkraft und der entsprechenden Auslenkung im Kontaktpunkt und im Bildpunkt.

Kontaktkraft $F_{2,z} \approx 8N$ entlang der Kontur geführt wird. Daraus ergibt sich in jedem Zeitschritt i eine Kraftdifferenz $\Delta F_{z,i} = F_{2z,i} - F_{1z,i}$ (siehe Abbildung 5.6(a) grün). Entsprechend ergibt sich die blau dargestellte Positions-differenz des Werkzeuges $\Delta s_{z,i}$ entlang der Kontur. Daraus wird die positionsabhängige Federkonstante im Kontaktpunkt berechnet (siehe Abbildung 5.6(b) blau):

$$K(l_i) = \frac{\Delta F_{z,i}}{\Delta s_{z,i}} \quad (5.7)$$

Die Deformation $\Delta \mathbf{p}$ nach Gleichung 5.6 beschreibt die Deformation des Werkstückes im aktuellen **Kontaktpunkt**. Zur Kompensation der Verformung des aufgezeichneten Pfades aus Kamerainformationen ist jedoch die Deformation im **Bildpunkt** von Bedeutung. Gleichung 5.6 gilt daher nur für die globale Verformung des Werkstückes, da in diesem Fall die Auslenkungen im Kontaktpunkt und im Bildpunkt identisch wären. Daher wird zur Berücksichtigung der lokalen Verformung bei der Berechnung

der Steifigkeit nicht die Positionsdivergenz Δs_z im Kontaktpunkt, sondern die Positionsdivergenz $\Delta s_{cam,z}$ der Bildinformationen ermittelt (siehe Abbildung 5.6(a) rot). Daraus wird die Steifigkeit

$$K_{cam}(l_i) = \frac{\Delta F_{z,i}}{\Delta s_{cam,z,i}} \quad (5.8)$$

im Bildpunkt der Kamera definiert²⁾.

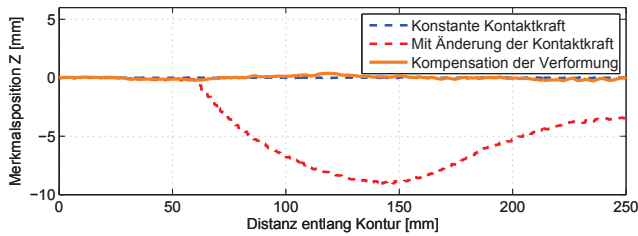
Der Versuchsaufbau besteht aus einem einseitig eingespanntem Metallblech (siehe Abbildung 5.2). Die Steifigkeit ist in der Mitte am höchsten. Das Metallblech wird durch den Kontakt nicht rein global verformt, d.h., durch Torsion ist die Auslenkung im Kontaktpunkt und im Bildpunkt der Kamera unterschiedlich (lokale Verformung). Dieser Effekt spiegelt sich im Vergleich der Federsteifigkeiten $K_{cam}(l_i)$ und $K(l_i)$ wider (siehe Abbildung 5.6(b)). Zur Kompensation der Bilddaten unter Berücksichtigung der lokalen Verformung wird Gleichung 5.6 folgendermaßen modifiziert:

$$\Delta \mathbf{p}_i^{TCP,i} = \begin{pmatrix} 0 & 0 & \frac{F_{z,i}}{K_{cam}(l_i)} \end{pmatrix}^T. \quad (5.9)$$

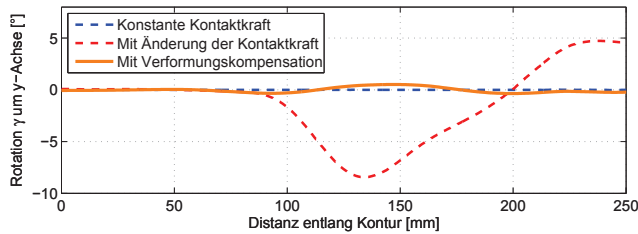
Die Kompensation der Objektverformung bei veränderter Kontaktkraft (siehe Abbildung 5.4(a)) ist in Abbildung 5.7(a) dargestellt. Die rote Linie zeigt das aufgezeichnete Bildmerkmal, während die Kontaktkraft verändert wird. Die orange Linie zeigt die Berechnung des unverformten Pfades nach Gleichung 5.5 und 5.9. Im Vergleich mit dem tatsächlich unverformten Pfad ohne Kontakt (blau) ergibt sich ein maximaler Fehler des kompensierten Pfades von 0,5mm zum unverformten Pfad.

Wird die Ausrichtung des Werkzeuges an den durch Kontakt verformten Pfad angepasst (rot), dann ergeben sich Orientierungsänderungen von bis zu 9° (siehe Abbildung 5.7(b) rot). Da jedoch ein großer Anteil der Verformung auf globale Verformung des Werkstückes zurückzuführen ist, sollte sich entlang des eigentlich geradlinigen Merkmales keine signifikante Orientierungsänderung bei wechselnder Kontaktkraft ergeben. Wird die Orientierung des Werkzeuges an die kompensierte Bildinformation (orange) angepasst, dann ergibt sich ein maximaler Orientierungsfehler von 0,5°. Der Orientierungsfehler wird entlang dieser Kontur somit um 95% reduziert.

²⁾Diese Definition der Steifigkeit gilt, solange bei aufeinanderfolgenden Ausführungen der Konturverfolgung die Vorlauflänge der Kamera einen annähernd gleichen Verlauf zeigt, da die Kraftmessung nicht im Bildpunkt erfolgt.



(a) Kompensation der Bildinformation bei Kontaktkraftänderung.



(b) Werkzeugorientierung mit und ohne Kompensation.

Abbildung 5.7: Durch Änderung der Kontaktkraft wird der Pfad verformt (vgl. Abbildung 5.4). Die Kompensation der Verformung reduziert den Orientierungsfehler entlang der Kontur um 95%.

Die Problematik der Orientierungsanpassung an die kompensierten Bildinformationen ist jedoch, dass eine tatsächliche Neigung des Pfades (aufgrund von z.B. Torsion des gesamten Werkstückes) nicht berücksichtigt wird. Diese Problematik ist in Abbildung 5.8 dargestellt. Die Anpassung des Werkzeuges z.B. senkrecht zum kompensierten Pfad erzeugt den Winkelfehler ϑ zum tatsächlichen Pfad. D.h., eine Anpassung der Orientierung um den Winkel ϑ um die y -Achse des Werkzeugkoordinatensystems ist notwendig. Durch Linearisierung der lokalen Verformung kann ϑ folgendermaßen genähert werden:

$$\sin \vartheta = \frac{\Delta s_z - \Delta s_{cam,z}}{l_{cam}} = \frac{\frac{F_z}{K} - \frac{F_z}{K_{cam}}}{l_{cam}} = \frac{F_z}{l_{cam}} \cdot \frac{K_{cam} - K}{K_{cam} \cdot K}, \quad (5.10)$$

mit l_{cam} der aktuellen Vorlauflänge der Kamera.

Die Berechnung des Winkels ϑ entlang des einseitig eingespannten Metallbleches bei einer Kontaktkraft von $F_z \approx 2N$ ist in Abbildung 5.9 darge-

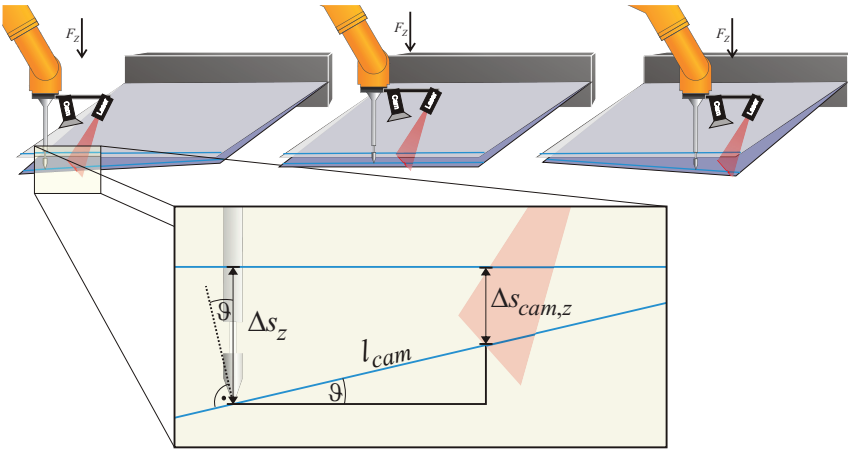
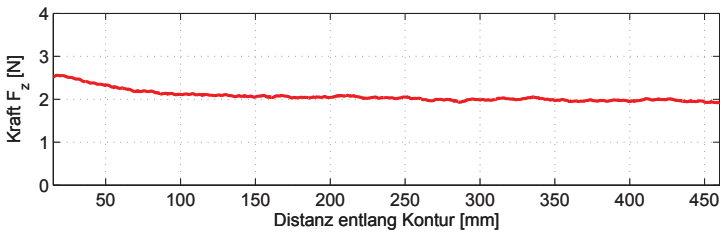
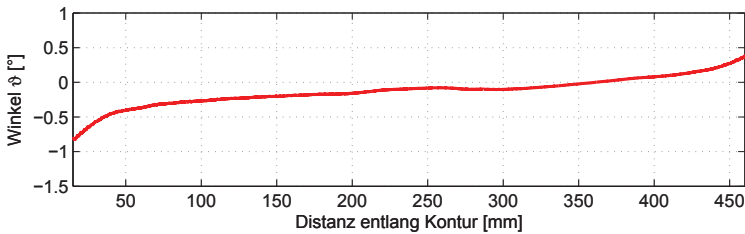


Abbildung 5.8: Bei Kontakt neigt sich die Kontur des Werkstücks. Die Werkzeugorientierung muss an den Winkel ϑ angepasst werden.



(a) Kontaktkraft entlang der Kontur.



(b) Neigungswinkel ϑ durch Kontaktkraft.

Abbildung 5.9: Neigung der Kontur durch Kontakt entlang des einseitig eingespannten Metallbleches.

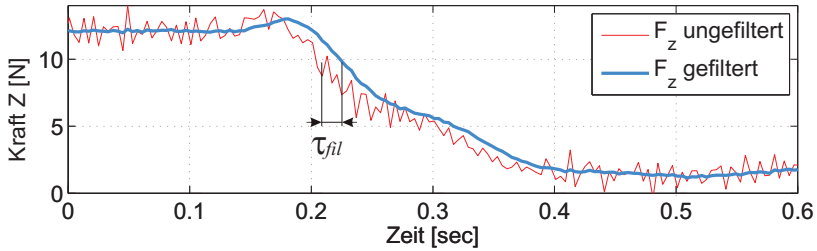


Abbildung 5.10: Die Kraftmessung an einem bewegten Roboter beinhaltet signifikantes Rauschen (hier mit einer Amplitude von $\pm 1N$). Die Glättung durch Filterung des Signales erzeugt eine Zeitverzögerung τ_{fil} .

stellt. Durch die Addition von ϑ auf die aktuelle Rotation um die y -Achse des Werkzeugkoordinatensystems wird der tatsächliche Fehler zum Werkstück reduziert.

5.2.2 Filterung und Zeitverzögerung zur Kompensation

Die Kompensation der Werkstückverformung durch Kontakt wird aus der Kraftmessung des KMS berechnet. Wie in Abbildung 5.10 dargestellt, beinhaltet die Kraftmessung an einem bewegten Roboter einen signifikanten Rauschanteil und sollte einer Filterung unterzogen werden. Störungen im Kraftsignal würden ansonsten die Bilddaten über die Kompensation negativ beeinflussen³⁾. Da die Filterung eine Verzögerung von τ_{fil} verursacht, ist die gefilterte Kraftmessung nicht mehr synchron mit dem Datenstrom der Kamera. Bei der Subtraktion der berechneten Verformung vom Bildsignal resultiert bei schneller Kraftänderung die Verzögerung in einer fehlerhaften Kompensation (siehe Abbildung 5.11 rot gestrichelt). Zur Synchronisation der Datenströme müssen die Bilddaten um die Zeit

$$\tau_{cam} = \tau_{fil} - \tau_{IP} \quad (5.11)$$

verzögert werden. τ_{IP} ist die Zeitverzögerung der Bildverarbeitung. Die explizite Verzögerung der Bilddaten um τ_{cam} ist unproblematisch für die

³⁾Bei einer Steifigkeit von $K = \frac{1N}{mm}$ würde bei Kompensation nach Gleichung 5.5 das Signalrauschen mit einer Amplitude von $\pm 1N$ ein zusätzliches Rauschen der kompensierten Bilddaten um $\pm 1mm$ erzeugen.

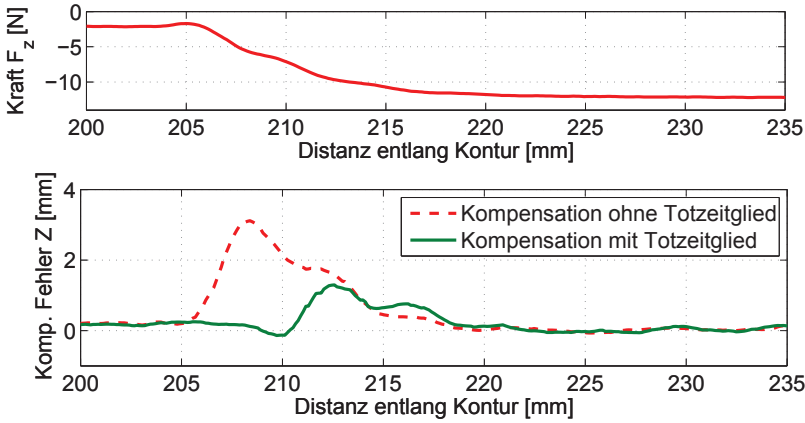


Abbildung 5.11: Bei schneller Änderung der Kontaktkraft F_z entsteht ein Kompensationsfehler aufgrund von Filterverzögerungen (rot gestrichelt). Der Kompensationsfehler wird durch den Einsatz eines Totzeitgliedes zur Kompensation der Filterverzögerung reduziert (grün).

bildgestützte Konturverfolgung, da die Daten im Vorlauf ermittelt werden. Somit ergibt sich lediglich in Abhängigkeit der aktuellen Geschwindigkeit v_x eine Verkürzung des nutzbaren Kameravorlaufs l_{adv} :

$$l_{adv}^* = l_{adv} - v_x \cdot \tau_{cam} \quad . \quad (5.12)$$

So würde bei einer Bewegungsgeschwindigkeit von z.B. $v_x = 50 \frac{mm}{sec}$ der Vorlaufbereich l_{adv} der Kamera bei einer Filterverzögerung von $40ms$ um lediglich $2mm$ verkürzt werden.

Wird die Bildinformation aus der Kamera mit einem Totzeitglied explizit verzögert, sodass die Bilddaten mit der gefilterten Kraftmessung synchron sind, ergibt sich ein geringerer Fehler der Kompensation (siehe Abbildung 5.11 grün). Der verbleibende Kompensationsfehler ist auf dynamische Effekte zurückzuführen und wird im nachfolgenden Abschnitt betrachtet.

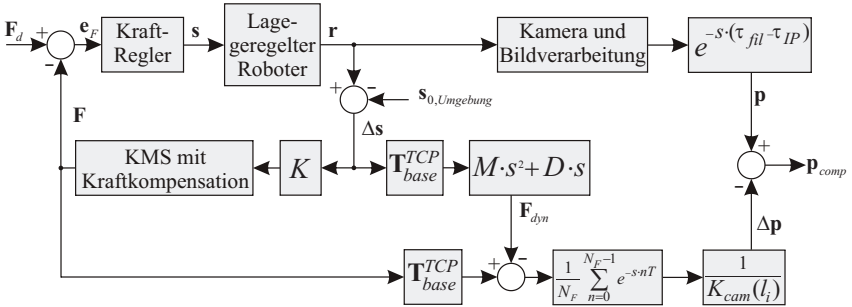


Abbildung 5.12: Die Objektverformung $\Delta \mathbf{p}$ wird über die Kontaktkraft und die Federsteifigkeit K_{cam} der Umgebung berechnet und von den gemessenen Bildmerkmalen \mathbf{p} zur Kompensation subtrahiert. Die aus der Filterung (gleitender Mittelwertfilter der Länge N_F) resultierende Zeitverzögerung und die Trägheits- und Reibungskräfte werden berücksichtigt.

5.2.3 Dynamische Kompensation der Kontaktkraft

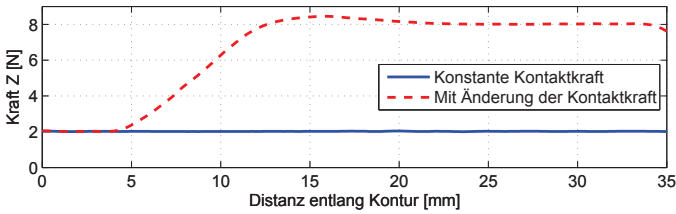
Durch die statische Kompensation wurde in Abschnitt 5.2.1 der Fehler der Werkzeugorientierung zur verfolgten Kontur reduziert. In den Experimenten wurde die Kraft relativ langsam verändert, sodass die vorgestellte statische Modellierung mittels der Federsteifigkeit ausreichend erschien. Bei Bewegungen mit höherer Dynamik müssen jedoch auch die Geschwindigkeit $\Delta \dot{s}_z$ und die Beschleunigung $\Delta \ddot{s}_z$ in Kontakttrichtung betrachtet werden. Die Kraft in Kontakttrichtung lässt sich durch den folgenden Zusammenhang eines Feder-Masse-Dämpfer Systems beschreiben:

$$F_z = M \cdot \Delta \ddot{s}_z + D \cdot \Delta \dot{s}_z + K \cdot \Delta s_z \quad , \quad (5.13)$$

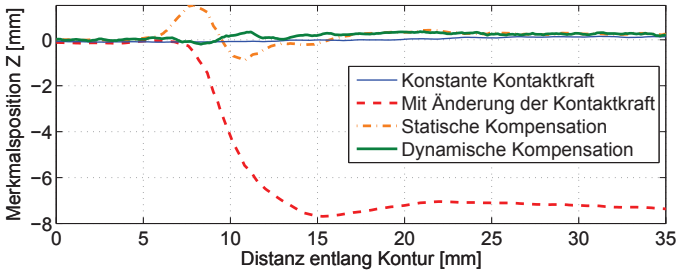
mit der Masse M , der Dämpfung D und der Steifigkeit K . Die Kräfte, welche durch Trägheit und Reibung verursacht werden, müssen von der gemessenen Kraft F_z subtrahiert werden, da diese Kraftkomponenten keine Verformung des Merkmals erzeugen. Die Verformung des Werkstückes in Werkzeugkoordinaten nach Gleichung 5.9 erweitert sich zu:

$$\Delta \mathbf{p}_i^{TCP,i} = \left(0 \quad 0 \quad \frac{F_{z,i} - M(l_i) \cdot \Delta \ddot{s}_{z,i} - D(l_i) \cdot \Delta \dot{s}_{z,i}}{K_{cam}(l_i)} \right)^T \quad . \quad (5.14)$$

Abbildung 5.12 zeigt das Blockschaltbild zur Kompensation mit Einbeziehung der dynamischen Effekte und der Filterung der Daten mit Be-



(a) Kontaktkraft bei Konturverfolgung



(b) Vergleich statischer und dynamischer Kompensation.

Abbildung 5.13: Kompensation der Objektverformung mit Berücksichtigung der Dynamik bei Änderung der Kontaktkraft.

rücksichtigung der Filterverzögerung zur Kompensation. Die Kompensation der Werkstückverformung unter Berücksichtigung der Parameter M und D ist in Abbildung 5.13 dargestellt. In diesem Experiment wurde die Kraft innerhalb kurzer Zeit von $2N$ auf $8N$ erhöht. Die rote Linie zeigt die unkompensierte Verformung der Kontur bei Änderung der Kontaktkraft. Die orange Linie zeigt die statische Kompensation (dabei wird ausschließlich die Steifigkeit des Materials berücksichtigt). Es wird deutlich, dass im Bereich der gleichbleibenden Verformung die statische Kompensation ausreichend ist. Während der Veränderung der Kraft ergibt sich jedoch eine deutliche Über- bzw. Unterkompensation. Wird die Dynamik mit berücksichtigt, ergibt sich die in grün dargestellte Kompensation, welche den unverformten Pfad mit einem Fehler von unter $0,5mm$ rekonstruiert, wohingegen die statische Kompensation einen Fehler von $1,5mm$ aufweist.

Die Ermittlung der zur dynamischen Kompensation notwendigen Parameter wird im nachfolgenden Abschnitt betrachtet.

5.2.4 Identifikation der Umgebung als Feder-Masse-Dämpfer System

Zur dynamischen Kompensation der Werkstückverformung ist neben der Kenntnis der Steifigkeit K auch die Kenntnis der bewegten Masse M und der Dämpfung D des Materials in Abhängigkeit der Position entlang der Kontur erforderlich. Diese Parameter können z.B. aus einem CAD Modell entnommen werden. Liegen diese Informationen nicht oder nur teilweise vor, können die Parameter über eine Identifikation vollständig ermittelt oder präzisiert werden. In diesem Abschnitt wird die Identifikation der Parameter M , D und K vorgestellt.

Die in Abschnitt 5.2.1 vorgestellte Ermittlung der Federsteifigkeit K basiert auf der Messung der Kraftdifferenz und der Positionsdifferenz entlang der Kontur. Die Abtastrate des Messsystems beeinflusst die Ermittlung dieses statischen Parameters nicht. Zur Ermittlung der dynamischen Parameter Masse M und Dämpfung D ist jedoch eine hohe Abtastrate der Sensorik notwendig, um Position und Kraft in Schwingungsversuchen oder Sprungantworten des Systems zu ermitteln. Damit scheidet die Kamera des vorhandenen Experimentalsystems als Messsystem aus, da die Abtastrate von $50fps$ zu gering ist, um dynamische Vorgänge aufzuzeichnen. Deshalb werden zur Ermittlung der notwendigen Parameter die Kraftmessung des KMS und die Positionsmessung des Roboters verwendet.

Eine vom Kamerasystem unabhängige Methode ist die Identifikation nach der Methode der kleinsten Quadrate für dynamische Prozesse [Ise92]. Dazu wird das Modell der Umgebung zwischen Eingang $U = F_z$ (Kraft) und Ausgang $Y = s_z$ (Auslenkung) folgendermaßen definiert:

$$G(s) = \frac{Y(s)}{U(s)} = \frac{1}{M \cdot s^2 + D \cdot s + K} \quad . \quad (5.15)$$

Die Diskretisierung des Modells nach Gleichung 5.15 ergibt das folgende, nicht sprungfähige, stabile und zeitinvariante System zweiter Ordnung mit der Totzeit d (d ist ein ganzzahliger Wert der Totzeit des abgetasteten Systems, welcher experimentell aus Messwerten bestimmt werden kann):

$$G(z) = \frac{y(z)}{u(z)} = \frac{B(z^{-1})}{A(z^{-1})} z^{-d} = \frac{b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}} z^{-d} \quad . \quad (5.16)$$

Die Identifikation nach der Methode der kleinsten Fehlerquadrate ist in Anhang C dargestellt. Daraus ergeben sich die Schätzwerte der Koeffizi-

enten \hat{b}_2 , \hat{b}_1 , \hat{a}_2 und \hat{a}_1 des diskreten Systems. Aus den Koeffizienten und der Abtastzeit T werden die Parameter des Modells im Frequenzbereich nach Gleichung 5.15 folgendermaßen berechnet (siehe Anhang D):

$$\hat{M} = -\frac{1}{4} \cdot \frac{T^2 \cdot (\hat{a}_1 - 1 - \hat{a}_2)}{\hat{b}_1 + \hat{b}_2} \quad (5.17)$$

$$\hat{D} = -\frac{T \cdot (\hat{a}_2 - 1)}{\hat{b}_1 + \hat{b}_2} \quad (5.18)$$

$$\hat{K} = \frac{1 + \hat{a}_2 + \hat{a}_1}{\hat{b}_1 + \hat{b}_2} \quad (5.19)$$

Die Gleichungen 5.17 bis 5.19 liefern Schätzwerte eines Systems zweiter Ordnung. Die Parameter sind jedoch nicht konstant über das gesamte Werkstück. Um die Parameter in Abhängigkeit der Position zu identifizieren, wird die Kontur in ein Raster unterteilt. Innerhalb der Blöcke dieses Rasters werden die Parameter unabhängig voneinander identifiziert. D.h., bei einer Unterteilung der Kontur in z.B. 50 Blöcke ergeben sich 50 unabhängige Parametermengen eines Feder-Masse-Dämpfer Systems in Abhängigkeit der Position entlang der Kontur.

Die experimentelle Ermittlung ist in Abbildung 5.14 dargestellt. Das Werkzeug wird entlang der Kontur geführt – dabei wird das Werkstück in Kontakttrichtung angeregt, um für die Identifikation die benötigten Eingangs- und Ausgangswerte des Systems zu erzeugen. Dazu können auch im System vorhandene Störungen genutzt werden (z.B. die Schwingungen einer robotergeführten Nähmaschine). In diesem Experiment wird die Anregung explizit mittels eines PRBS⁴⁾ Signales erzeugt. Abbildung 5.14 (oben) zeigt die Signale zur Identifikation entlang des einseitig eingespannten Metallbleches. Die Kontur wird in Blöcke mit einer Länge von 10mm unterteilt, sodass entlang der 450mm langen Kontur 45 unabhängige Systeme identifiziert werden. In Abbildung 5.14 (unten) sind am Beispiel des Blockes zwischen 340mm und 350mm die Signale vergrößert dargestellt.

Abbildung 5.15 zeigt den Verlauf von Federsteifigkeit, Masse und Dämpfung entlang der Kontur. Die Federsteifigkeit K entspricht dem Verlauf der zuvor in Abschnitt 5.2.1 ermittelten Federsteifigkeit. Die Anzahl der Durchläufe, die zur Identifikation der Parameter notwendig sind, hängt von der gewählten Blockbreite und von der Bewegungsgeschwindigkeit entlang der Kontur sowie von den Eingangs- und Ausgangssignalen ab. Bei einer

⁴⁾PRBS - Pseudorandom binary sequence

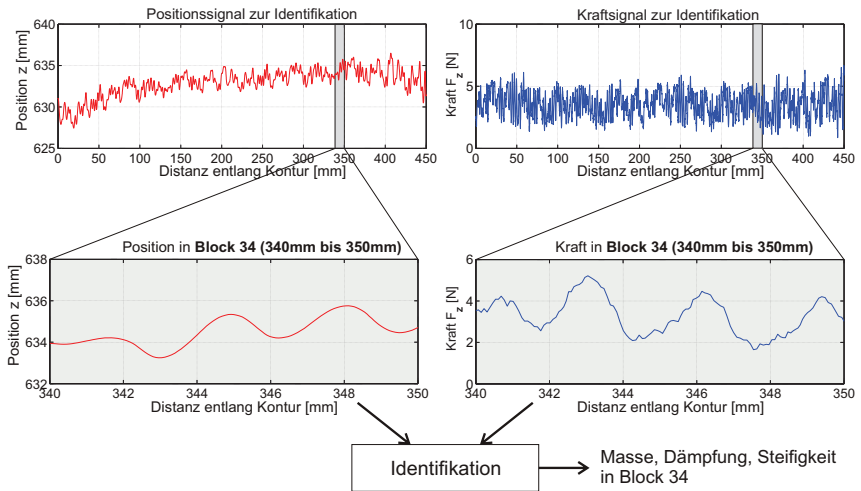


Abbildung 5.14: Eingangs- zu Ausgangsverhalten bei Anregung des Systems mit einem PRBS-Signal der Kraft. Die Kontur wird in Blöcke unterteilt, sodass die Parameter positionsabhängig bestimmt werden können.

Blockbreite von 10mm und einer Verfahrgeschwindigkeit von 20mm/sec wird jeder Block für $0,5\text{sec}$ pro Durchlauf angeregt. Für die Ermittlung der Parameter in Abbildung 5.15 wurden dabei ca. 15 Durchläufe benötigt, um die vollständig unbekannten Parameter zu identifizieren.

Das Positionssignal aus der Robotersteuerung wird durch Quantisierungsrauschen und Fehler durch Verzögerungen im Antriebsstrang negativ beeinflusst. Eine Verbesserung der Identifikation wird erreicht, indem die Qualität des Positionssignals erhöht wird. In Kapitel 2 wurde der Rigid-Body-Observer zur verbesserten Ermittlung der Roboterposition vorgestellt. In Abbildung 5.16 ist zu erkennen, dass die Federsteifigkeit während der Identifikation deutlich schneller einen stabilen Wert erreicht, wenn das Roboterpositionssignal aus dem Beobachter stammt.

In diesem Abschnitt wurde die Identifikation vorgestellt, um Parameter des Werkstückes während der Konturverfolgung online zu identifizieren. Auf Basis dieser Parameter kann die im vorangegangenen Abschnitt beschriebene dynamische Kompensation der Werkstückverformung durchgeführt werden. Diese Kompensation sorgt dafür, dass die bildgestützte

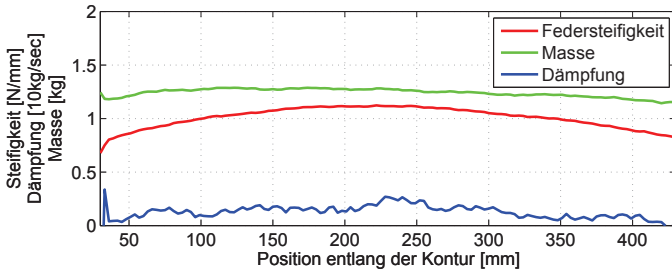


Abbildung 5.15: Identifizierte Parameter für Masse, Dämpfung und Steifigkeit in Abhängigkeit der Position entlang der Kontur.

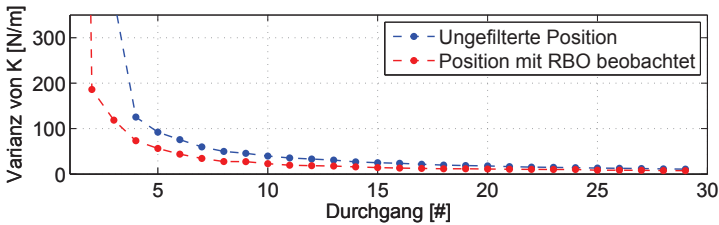


Abbildung 5.16: Die Identifikation erreicht deutlich schneller einen stabilen Wert, wenn die Roboterposition über einen Beobachter (RBO) gefiltert wird.

Konturverfolgung weitestgehend von Einflüssen durch wechselnde Kontaktkräfte entkoppelt betrachtet werden kann. Im folgenden Abschnitt wird die parallele Kraft- und bildgestützte Regelung vorgestellt und in Experimenten der Einfluss der vorgestellten Entkopplung gezeigt.

5.3 Gleichzeitige Kraft- und Bildregelung

In den vorangegangenen Kapiteln wurden die Kraftregelung, die bildgestützte Konturverfolgung und die Entkopplung dieser beiden Regelkreise vorgestellt. Es wurde bereits experimentell gezeigt, dass die bildgestützte Anpassung der **Orientierung** durch die Kompensation der Werkstückverformung von Änderungen der Kontaktkraft an nachgiebigen Werkstücken

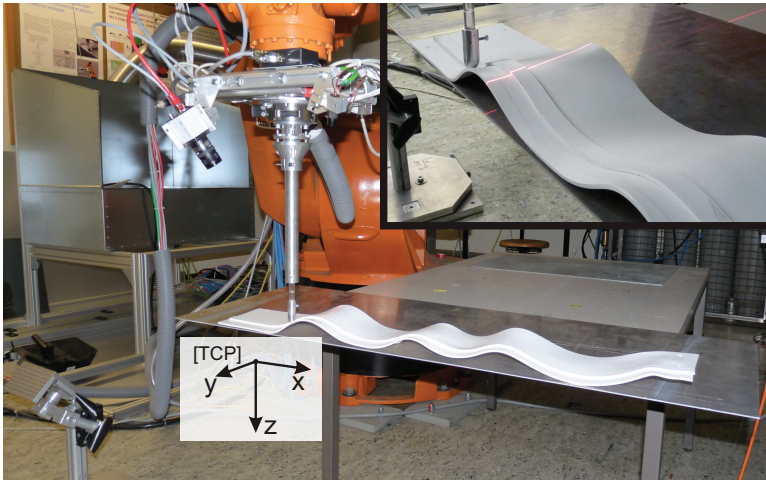


Abbildung 5.17: Foto eines Wellenprofils. Die Kontur auf dem Profil wird durch bildgestützte Regelung verfolgt. Durch Kraftregelung wird die gewünschte Kontaktkraft hergestellt.

entkoppelt wird. In diesem Abschnitt wird die Entkopplung der **translatorischen** Bewegung von Einflüssen durch die Kontaktkraft in den folgenden zwei Experimenten untersucht:

- Vorsteuerung des Kraftreglers aus Bildinformationen.
- Gleichzeitige translatorische bildgestützte Regelung und Kraftregelung in einer Achsrichtung.

5.3.1 Experiment 1: Kraftvorsteuerung aus Bildinformationen

In diesem Experiment wird die Kontur eines Wellenprofils verfolgt. Der Versuchsaufbau ist in Abbildung 5.17 dargestellt. Das Ziel ist, während der Bewegung entlang der Kontur eine konstante Kontaktkraft F_z in z -Richtung des Werkzeugkoordinatensystems $K_{[TCP]}$ einzuhalten. Die vollständige Aufgabendefinition ist in Tabelle 5.1 zusammengefasst.

Tabelle 5.1: Aufgabendefinition Experiment 1. Die Bewegung wird im Werkzeugkoordinatensystem $K_{[TCP]}$ vorgegeben.

| | |
|------------|--|
| x : | Bahngeschwindigkeit $v_x = 20 \frac{mm}{sec}$ entlang der Kontur. |
| y : | Abstand $d_y = 0mm$ des Werkzeuges zur Kontur. |
| z : | Kraft $F_z = 2N$ zwischen Werkzeug und Kontur. |
| γ : | Geschwindigkeit um x -Achse $\frac{d\gamma}{dt} = 0$. |
| β : | Geschwindigkeit um y -Achse $\frac{d\beta}{dt} = 0$. |
| α : | Rotation um z -Achse, sodass die x -Achse tangential zur Kontur liegt. |

Berechnung der Kraftvorsteuerung

Der Fokus dieses Experiments liegt auf der Kraftregelung zur Einhaltung der gewünschten Kontaktkraft F_z . Die Kraftregelung wird in dieser Arbeit durch die Admittanzregelung realisiert (siehe Kapitel 4), d.h., dem Regelfehler \mathbf{e}_F wird eine Geschwindigkeit des Werkzeugkoordinatensystems zugeordnet. Die Umsetzung dieser Geschwindigkeitsvorgabe wird durch den internen Lageregler des Roboters realisiert.

Durch den unterlagerten Lageregler entstehen Verzögerungen im Regelkreis, sodass bereits bei geringen Störgrößen Instabilität auftritt, falls die Reglerparameter des Kraftreglers mit zu großen Verstärkungsfaktoren ausgelegt werden. Hingegen wird Stabilität erreicht, wenn die Reglerparameter mit verhältnismäßig niedrigen Verstärkungsfaktoren ausgelegt werden. Dadurch kann jedoch nur langsam auf Störungen reagiert werden, und es können große Regelfehler entstehen. Wird die Bewegung des Roboters jedoch durch die Schätzung der erwarteten Störgrößen vorgesteuert, kann der Regelfehler wiederum reduziert werden. Diese Vorsteuerung wird mit der Vorsteuergeschwindigkeit $v_{ff,z}$ in z -Richtung des Werkzeugkoordinatensystems realisiert. Das Blockschaltbild der Kraftregelung mit Vorsteuerung ist in Abbildung 5.18 dargestellt.

Die Störgröße bei kraft geregelter Konturverfolgung ist in diesem Experiment die Veränderung der z -Position der Kontur. Aus Bildinformationen kann diese Veränderung ermittelt und zur Vorsteuerung des Roboters ver-

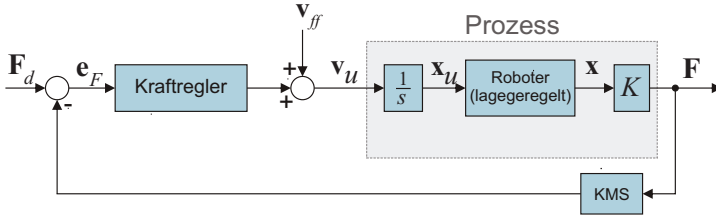


Abbildung 5.18: Kraftregelung mit Geschwindigkeitsvorsteuerung. Die Vorsteuerung mit der Geschwindigkeit \mathbf{v}_{ff} reduziert den Regelfehler.

wendet werden. Die Berechnung der Vorsteuerung wird in Abbildung 5.19 gezeigt. Die Veränderung Δz des Merkmals innerhalb des Vorlaufbereiches wird aus dem Merkmalspfad \mathbf{P} ermittelt, indem der Abstand in z -Richtung des Werkzeugkoordinatensystems zwischen der aktuellen Position⁵⁾ \mathbf{p}_{act} und einer vorausliegenden Position \mathbf{p}_{adv} berechnet wird. Dabei hat \mathbf{p}_{adv} den Abstand s_{xy} (z.B. 10mm) zu \mathbf{p}_{act} in der x - y -Ebene des Werkzeugkoordinatensystems. Die Vorlaufzeit

$$t_{adv} = \frac{s_{xy}}{v_{xy}} \quad (5.20)$$

für die Bewegung zwischen \mathbf{p}_{act} bis zum Vorlaufpunkt \mathbf{p}_{adv} wird aus der aktuellen Geschwindigkeit v_{xy} des Roboters in der x - y -Ebene des Werkzeugkoordinatensystems berechnet. Daraus ergibt sich die Vorsteuergeschwindigkeit

$$v_{ff,z} = \frac{\Delta z}{t_{adv}} \quad (5.21)$$

in z -Richtung des Werkzeugkoordinatensystems.

Die Kraftregelung mit Vorsteuerung wird am Beispiel der in Abbildung 5.20(a) dargestellten Kontur untersucht. Die Kontaktkraft F_z , die durch Kraftregelung **ohne Vorsteuerung** entlang der Kontur auftritt, ist in Abbildung 5.20(b) (blau) dargestellt. Durch die Krümmung der Kontur in z -Richtung entstehen deutliche Abweichungen vom Sollwert $F_{d,z} = 2\text{N}$, da die Änderung der z -Position entlang der Kontur ausschließlich durch die Kraftregelung berücksichtigt wird. Unter Verwendung der

⁵⁾Der Zusammenhang zwischen der aktuellen Roboterposition \mathbf{r}_{act} zum korrespondierenden Punkt \mathbf{p}_{act} im Merkmalspfad wurde in Abschnitt 3.2.1 vorgestellt.

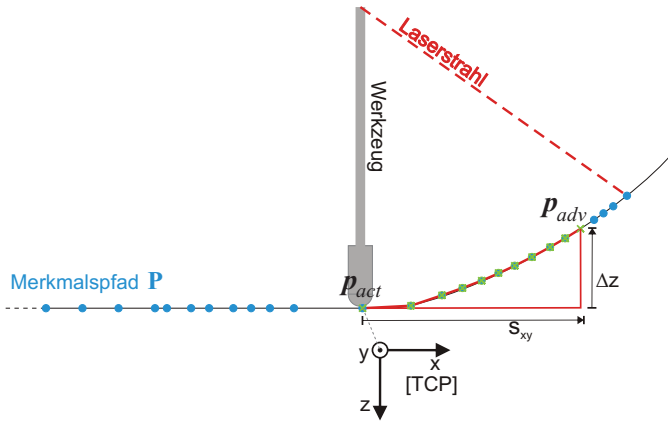
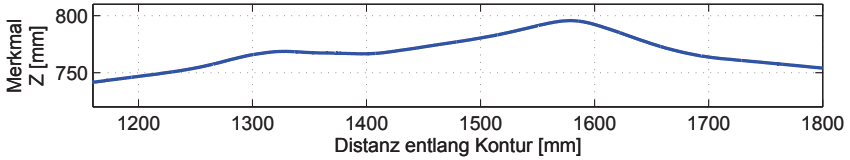


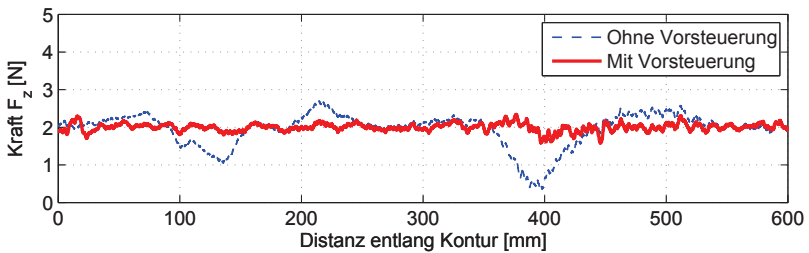
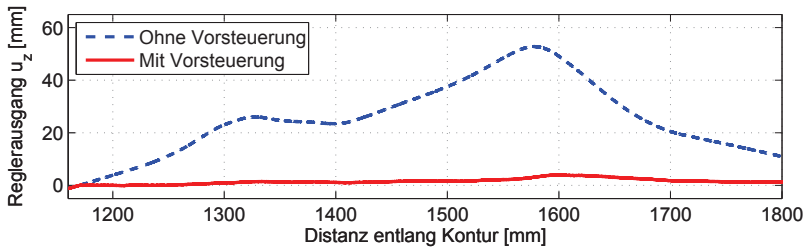
Abbildung 5.19: Berechnung der Vorsteuerung. Zum aktuellen Punkt p_{act} wird im Abstand s_{xy} der Abstand Δz in z-Richtung des Werkzeugkoordinatensystems $K_{[TCP]}$ berechnet und zur Vorsteuerung verwendet. Die Ermittlung von p_{act} wurde in Abschnitt 3.2.1 vorgestellt.

Kraftregelung **mit Vorsteuerung** wird der Regelfehler der Kraft deutlich reduziert (rot). Durch die Vorsteuerung wird der notwendige Einfluss der Kraftregelung auf die z-Position des Werkzeuges geringer, sodass lediglich die Ungenauigkeiten der Vorsteuerung und weitere Störgrößen mittels Kraftregelung berücksichtigt werden müssen (Abbildung 5.20(c) (rot). Ohne Vorsteuerung ist der Reglerausgang des Kraftreglers deutlich größer (blau).

Bei Kraftregelung mit Vorsteuerung durch Bildinformation ist die Entkopplung der Regelkreise durch Kompensation der Werkstückverformung notwendig. Wird während der Konturverfolgung die gewünschte Kontaktkraft von $F_{d,z} = 12N$ auf $F_{d,z} = 2N$ reduziert, entstehen Störungen durch die Bildverarbeitung. Diese Störungen entstehen dadurch, dass eine Vorsteuergeschwindigkeit in Richtung der Verformung des Werkstückes aufgrund der Kontaktkraftänderung berechnet wird (siehe Abbildung 5.21 rot). Die gemessene Änderung der z-Position entspricht jedoch keiner Änderung der z-Position innerhalb des Werkstückes. Diese Störungen werden reduziert, wenn die Kompensation der Verformung angewendet wird – die Schwingungen nach der Kontaktkraftänderung werden dadurch deutlich reduziert (blau).



(a) Zu verfolgende Kontur auf einem Wellenprofil.

(b) Kontaktkraft in z -Richtung.

(c) Ausgang des Kraftreglers. Durch die Vorsteuerung ist nur ein geringer Einfluss des Kraftreglers notwendig (rot).

Abbildung 5.20: Kraftregelung mit Vorsteuerung entlang der Kontur eines Wellenprofils (a). Die Kontur soll in z -Richtung mit konstanter Kontaktkraft $F_{d,z} = 2\text{N}$ verfolgt werden. Die Vorsteuerung reduziert den Regelfehler der Kraftregelung erheblich.

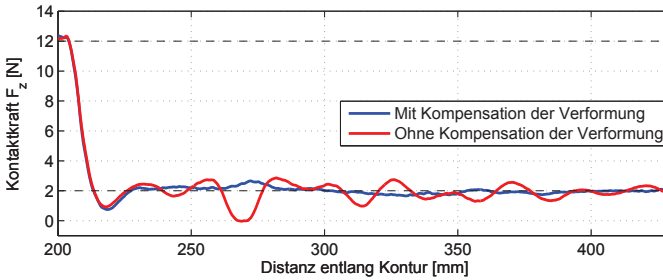


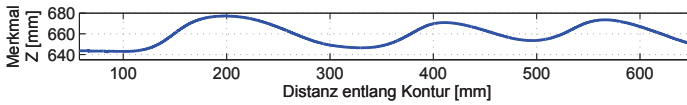
Abbildung 5.21: Die Änderung der Kontaktkraft während der Kraftregelung mit Vorsteuerung erzeugt deutliche Störungen (rot). Diese Störungen werden durch die Kompensation der Werkstückverformung deutlich reduziert (blau).

5.3.2 Experiment 2: Bildgestützte Regelung und Kraftregelung in einer Achsrichtung

In diesem Experiment wird die in Abbildung 5.22(a) dargestellte Kontur verfolgt. Die Position in z -Richtung des Werkzeugkoordinatensystems wird gleichzeitig durch eine Kraftregelung und eine bildgestützte Regelung beeinflusst. Diese Art der parallelen Regelung ist prinzipiell aufgrund der gegenseitigen Beeinflussung der Regelkreise nicht praktikabel, sie soll im Folgenden dennoch zeigen, dass auch für diesen Fall eine Entkopplung der Regelkreise möglich ist.

Zunächst wird der Abstand $d_z = 0\text{ mm}$ zwischen TCP und Kontur in z -Richtung rein bildgestützt geregelt. Dies bedeutet, dass bei idealer Kalibrierung des gesamten Systems der TCP exakt mit der Kontaktkraft $F_z = 0\text{ N}$ auf der Oberfläche geführt werden würde. Da jedoch immer ein minimaler Kalibrierungsfehler verbleibt, ergibt sich in diesem Experiment eine Kontaktkraft $F_z > 0\text{ N}$. Durch diese Kontaktkraft wird das Werkstück in z -Richtung verformt. Somit ergibt sich aus der Bildverarbeitung ein in z -Richtung verformter Pfad \mathbf{P} , an welchen wiederum die Position des TCP angepasst wird – dadurch wird das Werkzeug noch weiter in z -Richtung bewegt. D.h., es besteht eine Mitkopplung, sodass die Kontaktkraft stark ansteigt (siehe Abbildung 5.22(b) rot).

Wird nun einfach zu der bildgestützten Regelung ein Kraftregler parallel geschaltet, welcher die Kontaktkraft in z -Richtung regelt, können die



(a) Kontur auf dem Wellenprofil.

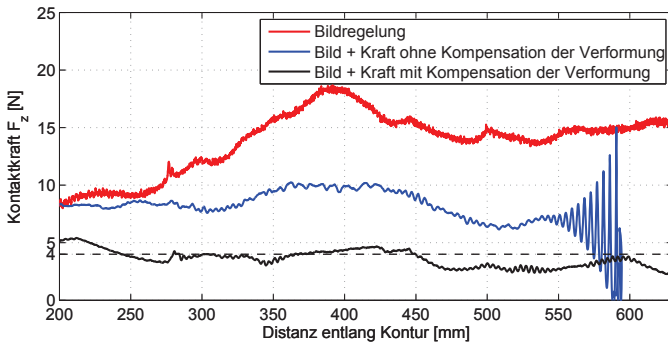
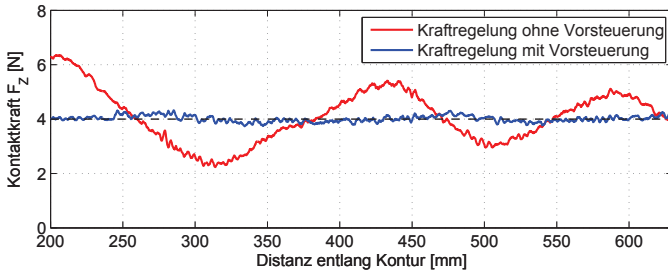
(b) Kontaktkraft bei $F_{d,z} = 4N$. Die gleichzeitige Kraftregelung und Bildregelung führt zu Schwingungen (blau); die Kompensation der Werkstückverformung reduziert die Verkopplung der Regelkreise (schwarz).(c) Kontaktkraft bei $F_{z,soll} = 4N$. Bei Verwendung von Kraftregelung ohne Vorsteuerung entsteht ein großer Regelfehler (rot). Die Vorsteuerung reduziert den Regelfehler deutlich (blau).

Abbildung 5.22: Die Kontur (a) soll mit konstanter Kontaktkraft verfolgt werden. Die parallele Verwendung von bildgestützter Regelung und Kraftregelung sorgt für Instabilität durch Verkopplung. Die Entkopplung durch Kompensation der Werkstückverformung reduziert die Schwingungen.

beiden Regelkreise stark aufschwingen, da sie über die Werkstückverformung verkoppelt sind (blau). Diese Verkopplung wird aufgehoben, wenn die Kompensation der Werkstückverformung angewendet wird (schwarz). Da die Modellierung der Werkstückparameter nicht zu 100% genau ist, wird jedoch die gewünschte Kontaktkraft von $F_z = 4N$ nicht exakt eingehalten, da die bildgestützte Regelung weiterhin minimal mit der Verformung des Werkstückes verkoppelt ist. Jedoch schwingen die Regelkreise nicht gegeneinander auf.

Wie zu Beginn erwähnt, ist diese Art der parallelen Regelung zur Konturverfolgung nicht praktikabel; es zeigt dennoch die Entkopplung durch die vorgestellte Kompensation der Werkstückverformung. Für das Experiment eignet sich die zuvor beschriebene Kraftregelung mit Vorsteuerung, um die gewünschte Kontaktkraft $F_{d,z} = 4N$ einhalten zu können (siehe Abbildung 5.22(c) blau). Eine reine Kraftregelung ist auch in diesem Experiment nicht ausreichend, um den Regelfehler adäquat zu minimieren (rot).

5.4 Zusammenfassung

Die gleichzeitige Regelung mit Kraft- und Bildinformationen zur Konturverfolgung wurde in diesem Kapitel betrachtet. Bei Kraftregelung, insbesondere wenn die Kontaktkraft oder die Steifigkeit entlang der Kontur nicht konstant sind, entstehen durch Kontakt an nachgiebigen Werkstücken Verformungen, welche sich entlang der Kontur verändern. Durch die Bildverarbeitung wird die dreidimensionale Position von Merkmalen entlang der Kontur während der Bewegung aufgezeichnet. Verändert sich die Verformung des Werkstückes während der Konturverfolgung, dann ergibt sich ein im dreidimensionalen Raum scheinbar verformter Merkmalspfad \mathbf{P} . Durch diese gemessene Verformung, welche nicht der tatsächlichen Form des Werkstückes im Kontaktpunkt entspricht, ergeben sich bei Anpassung der Orientierung Fehlstellungen des Werkzeuges zur Kontur. D.h., die bildgestützte Regelung ist mit den Verformungen, welche durch die Kraftregelung entstehen, verkoppelt.

Um diese Verkopplung zu lösen, wurde ein Modell der Umgebung als Feder-Masse-Dämpfer System mit positionsabhängigen Parametern beschrieben. Die Parameter wurden mittels online Identifikation aus Messwerten des Roboters und der Kraftregelung ermittelt. Die Identifikation

konnte weiterhin verbessert werden, wenn statt der ungefilterten Roboterposition die in Kapitel 2 vorgestellte Beobachtung der Roboterposition unter Einsatz von Beschleunigungssensoren verwendet wurde.

Des Weiteren wurde auf die Zeitverzögerung eingegangen, die entsteht, wenn die Signale aus dem KMS zur Glättung gefiltert werden, da dadurch die Datenströme aus Kamera und KMS nicht mehr synchron sind. Um diese Datenströme wieder zu synchronisieren, wurde die Messung der Bildinformationen entsprechend der entstandenen Filterverzögerung mittels Totzeitglied verzögert. Diese Verzögerung beeinflusst nicht die Stabilität der bildgestützten Regelung, sondern verkürzt lediglich den verwendbaren Vorlaufbereich der Kamera.

Im letzten Abschnitt dieses Kapitels wurden zwei Experimente zur Regelung der z -Position entlang einer Kontur vorgestellt. Es wurde die Kraftregelung mit Vorsteuerung aus Bildinformationen präsentiert und der Vorteil auf die Regelung durch Kompensation der Werkstückverformung gezeigt. Im letzten Experiment wurde deutlich, dass die Modellierung mittels linearem Feder-Masse-Dämpfer System keine 100%-ige Kompensation der Werkstückverformung ermöglicht, jedoch die Instabilität zwischen den Regelkreisen erfolgreich vermeidet.

6 Zusammenfassung und Ausblick

Der Trend zur Automatisierung von Fertigungsprozessen fördert die Verbreitung von Robotern in vielen Industriezweigen. Im Jahr 2011 waren weltweit schätzungsweise 1,1 Millionen Industrieroboter im aktiven Einsatz¹⁾. Trotzdem sind Robotersysteme auch heutzutage bei Kleinserienproduktionen unrentabel, da sie häufig mit wenig Sensoreinsatz verwendet werden. Dies fordert einen hohen Aufwand und Fachwissen beim Programmieren der benötigten Bewegung. Die Bewegung des Roboters, um z.B. Dichtmasse entlang einer Schweißnaht aufzutragen, ist eigentlich durch die Kontur des Werkstückes klar definiert – jedoch bedarf es einer gesteigerten Autonomie durch Einsatz von Sensoren, um Merkmale der Umgebung wahrzunehmen und zu interpretieren. Gesteigerte Autonomie erhöht folglich die Effizienz durch die Reduktion notwendiger Programmierung.

Diese Dissertation leistet einen Beitrag zur Erhöhung der Autonomie von Robotersystemen zur Konturverfolgung. Verfahren zur Regelung mit taktilen und visuellen Daten wurden vorgestellt. Dafür wurden zusätzlich die Messwerte aus Beschleunigungssensorik und aus der Positionsmessung des Roboters verwendet. Die Informationen dieser vier Sensorsysteme wurden in einer gemeinsamen Regelungsstruktur verbunden.

In Kapitel 3 wurde ein Regelungskonzept auf Basis eines vorausschauenden Laser-Triangulationssensors zur Konturverfolgung vorgestellt. Der Regelkreis wird im Gegensatz zum klassischen Visual Servoing nicht direkt über die Kamerainformation geschlossen, sondern über die internen Achsregler des Roboters (Dynamic Look-And-Move). Dieser Ansatz wurde gewählt, da einerseits das verwendete Robotersystem nur eine Schnittstelle zur Vorgabe kartesischer Sollwerte zur Verfügung stellt und andererseits eine kartesische Trajektorie vorausschauend berechnet wird.

¹⁾Quelle: International Federation of Robotics – <http://www.ifr.org/>

Es wurde ein Algorithmus zur Erfassung, Sortierung und Filterung von Bildmerkmalen entwickelt. Translatorische und rotatorische Stellwerte werden aus Bildinformationen generiert. Auch die Bewegungsgeschwindigkeit entlang der Kontur wird berücksichtigt, sodass der Pfad in Ecken exakt verfolgt wird und die Bewegungsgeschwindigkeit in engen Kurven an die Krümmung des Pfades angepasst wird. Durch die Prädiktion der Roboterbewegung entlang des Pfades auf Basis der aktuellen Geschwindigkeit und der Systemverzögerungen wird der Pfad des Roboters auch bei signifikanten Filterverzögerungen präzise an die Kontur angepasst. So wurde in Experimenten an Merkmalen mit starker Krümmung die Bewegungsgeschwindigkeit zur Minimierung des Orientierungsfehlers zwischen Werkzeug und Kontur automatisch angepasst. An Merkmalen mit geringen Krümmungen in der Ebene konnte bei Experimenten mit einer Geschwindigkeit von $600 \frac{mm}{sec}$ entlang einer vollständig unbekannten Kontur ein Positionsfehler von unter $1mm$ erreicht werden.

Die vorgestellte positionsbasierte bildgestützte Regelung ist stark von der Kalibrierung abhängig. Diese Abhängigkeit umfasst sowohl die exakte Kalibrierung des Werkzeugkoordinatensystems, als auch die exakte Kenntnis der aktuellen Kamerapose. Der Zusammenhang zwischen dem Kamerakoordinatensystem und dem TCP des Roboters ist durch Kalibrierung bekannt; der Zusammenhang zwischen dem Werkzeugkoordinatensystem und dem Basiskoordinatensystem wird aus der Vorwärtstransformation der Gelenkwinkel des Roboters berechnet. In Kapitel 2 wurde jedoch gezeigt, dass die Ermittlung der kartesischen Roboterposition durch Nachgiebigkeiten im Antriebsstrang nicht mit der tatsächlichen Position des TCP übereinstimmt – es wurde ein Fehler von mehreren Millimetern gemessen. Dieser Fehler wird bei der Umrechnung von Bildkoordinaten in Basiskoordinaten direkt auf die 3D-Bildmerkmale übertragen. Daher wurde ein Beobachter vorgestellt, um mithilfe von Beschleunigungssensoren die tatsächliche Position des TCP präziser zu erfassen.

Einige Anwendungen erfordern die Einhaltung einer definierten Kontaktkraft. Die Kontaktkraft wird mit einem Kraft-Momenten Sensor (KMS) gemessen. Wie in Kapitel 4 gezeigt, beinhaltet die Messung neben der Kontaktkraft auch Einflüsse durch Gravitation und Trägheit. Zur Berechnung der Trägheitskräfte wurde Beschleunigungssensorik eingesetzt. Besonders bei der Verwendung eines nachgiebigen KMS entstehen Bewegungen zwischen Roboterflansch und Werkzeug, welche durch die Kenntnis der berechneten Beschleunigung am Roboterflansch nicht erfasst werden können. Es wurde gezeigt, dass die Kompensation von Trägheitskräften deutlich

verbessert werden kann, wenn die Beschleunigung direkt am Werkzeug selbst gemessen wird, da dadurch die Bewegungen erfasst werden, welche die Trägheitskräfte in der Kraftmessung tatsächlich verursachen. Die Kraftkompensation ist somit von Trägheitskräften durch Beschleunigungen des Roboters entkoppelt – also auch von Beschleunigungen, welche durch bildgestützte Regelung vorgegeben werden.

Zur Regelung der Kontaktkraft wurde in Kapitel 4 die explizite positionsbasierte Kraftregelung vorgestellt, welche einem Regelfehler der Kraft einen Geschwindigkeitssollwert bzw. Lagesollwert für die internen Lage-regler des Roboters zuordnet. Durch den unterlagerten Lageregelkreis entsteht eine Verzögerung, welche kaum zu beeinflussen ist (die Filterverzögerung und Totzeit des Roboters wurde in Abschnitt 1.4 gezeigt). Durch diese Verzögerungen muss die Kraftregelung mit relativ geringen Verstärkungsfaktoren ausgelegt werden, um stabil zu bleiben. Die Auslegung mit geringen Verstärkungsfaktoren führt jedoch dazu, dass die Kraftregelung nur langsam auf Störgrößen reagieren kann, was zu den in Abschnitt 5.3.1 gezeigten Regelfehlern führt. Der KMS liefert zwar eine sehr genaue Messung der Kraft im Kontaktpunkt, durch die Verzögerungen kann jedoch nicht schnell genug darauf reagiert werden. Zur Lösung dieses Problems wurde der Nutzen der Kraftvorsteuerung aus Bildinformationen in Abschnitt 5.3.1 gezeigt. Das Geschwindigkeitssignal zur Vorsteuerung berücksichtigt auftretende Störgrößen, um den Regelfehler der Kraft zu reduzieren. Durch diese Fusion von Kraft- und Bildinformation müssen lediglich unvorhersehbare Störungen und Ungenauigkeiten der Vorsteuerung durch die Kraftregelung ausgeglichen werden, sodass der Regelfehler in Experimenten deutlich gesenkt werden konnte.

Im letzten Kapitel wurde die Fusion von Bild- und Kraftinformation betrachtet. Der Einfluss auf die bildgestützte Regelung durch die Verformung eines nachgiebigen Werkstückes bei Kontakt wurde untersucht. Die Verkopplung zwischen der bildgestützten Regelung und der Kontaktkraft wird gelöst, indem die Verformung des Materials über ein Modell der Umgebung berechnet wird, um die Verformung in den Bildmerkmalen zu kompensieren. An unterschiedlichen Beispielen wurde gezeigt, dass die bildgestützte Regelung von der Kraftregelung weitestgehend entkoppelt werden konnte.

Insgesamt hat diese Arbeit deutlich gezeigt, dass Verkopplungen bei gleichzeitiger Verwendung unterschiedlicher Sensorsysteme entstehen, welche kompensiert werden müssen. Allerdings ergeben sich durch die Fusion von unterschiedlichen Sensorsystemen signifikante Vorteile, wenn unter-

schiedliche Messmethoden zur Generierung eines Messwertes verwendet werden – so kann zur präzisen Bestimmung der Position des TCP die hohe stationäre Genauigkeit der Roboterposition (aus der Vorwärtstransformation) mit der hochgenauen Beschleunigungsmessung am TCP verbunden werden. Dadurch wird sowohl in dynamischen als auch in stationären Zuständen ein optimales Messergebnis erreicht. Des Weiteren wurden Nachteile durch die direkte Rückführung von Sensorinformationen in einem Regelkreis mit Totzeiten durch die Fusion von Kraft- und Bildinformation durch Vorsteuerung ausgeglichen.

Das Ergebnis dieser Arbeit bedeutet einen Fortschritt im Bereich der Konturverfolgung mit Industrierobotern. Die industrielle Relevanz der in dieser Arbeit vorgestellten Mechanismen wird dadurch verstärkt, dass die Algorithmen unabhängig von den verwendeten Systemkomponenten sind. D.h., die Vorgabe basiert auf einer für die meisten Industrieroboter verfügbaren Schnittstelle zur kartesischen Positionsvorgabe für die interne Lagereglung. Im Vergleich zu dem in der Literatur häufig verwendeten direkten Visual Servoing auf Gelenkebene ergeben sich durch die kartesische Vorgabe Probleme, welche jedoch in dieser Arbeit eingehend untersucht und gelöst wurden - auch mit Systemverzögerungen wird eine stabile und dennoch präzise Regelung hinsichtlich der Konturverfolgung erreicht. Besonders der Einsatz von Beschleunigungssensoren, welche heutzutage als Massenprodukt extrem günstig sind, hat die Regelung in mehreren Bereichen verbessert. Insbesondere der vorgestellte Beobachter stellt mit geringem Mehraufwand einen entscheidenden Fortschritt dar.

Eine zukünftige Weiterentwicklung der Multisensor-Konturverfolgung sollte im Bereich der Kompensation der Werkstückverformung geschehen. Die Modellierung als positionsabhängiges Feder-Masse-Dämpfer System ist für die meisten Materialien nicht ideal. Eine Modellierung z.B. mit der *Finite Elemente Methode* (FEM) lässt ein besseres Ergebnis erwarten.

Potenzial zur Weiterentwicklung liegt auch im Bereich der bildgestützten Konturverfolgung. Aktuell werden keine Singularitäten des Roboters und auch keine Gelenkgeschwindigkeiten in der Trajektorienplanung berücksichtigt. Die reine Überwachung der aktuellen Gelenkgeschwindigkeiten wäre unproblematisch – die vorausschauende Berechnung der auftretenden Gelenkgeschwindigkeiten kann jedoch nur auf Basis eines dynamischen Modells des Roboters erfolgen. Mit einem solchen Modell wäre es möglich, auch Gelenkgeschwindigkeiten im Voraus durch die Anpassung der Bewegungsgeschwindigkeit entlang der Kontur zu berücksichtigen.

A Dreidimensionale lineare Regression

Zur Approximation einer Tangente an reale Messwerte werden benachbarte Punkte des Tangentenpunktes zur Berechnung der Tangente mit einbezogen. Für die nachfolgend beschriebene lineare Regression werden alle Elemente zwischen \mathbf{p}_Q und \mathbf{p}_R zur Approximation verwendet. D.h., die lineare Regression wird auf ein Pfadsegment

$$\mathbf{p}_j \quad \text{mit } j = Q, \dots, R \quad (\text{A.1})$$

in der Umgebung des Vorlaufpunktes \mathbf{p}_{adv} angewendet. Die Merkmale \mathbf{p}_j werden auf eine Regressionsebene projiziert. Anschließend wird die lineare Regression in \mathbb{R}^2 berechnet (siehe Abbildung A.1). Die Regressionsebene \vec{E}_R wird in Abhängigkeit der Koordinaten $\mathbf{x} = (x, y, z)^T$ folgendermaßen definiert:

$$\vec{E}_R : \quad y = a + b \cdot x + c \cdot z \quad . \quad (\text{A.2})$$

Die unbekannten Parameter a, b, c werden mit der Methode der minimalen Fehlerquadrate aus den Messwerten $\mathbf{p}_j = (p_{xj}, p_{yj}, p_{zj})^T$ durch Minimierung des Gütekriteriums

$$r = \sum_{j=Q}^R [p_{yj} - (a + b \cdot p_{xj} + c \cdot p_{zj})]^2 \rightarrow \min \quad (\text{A.3})$$

ermittelt. Nullsetzen der partiellen Ableitung $\frac{\partial r}{\partial a}$ liefert den Parameter a :

$$\begin{aligned} \frac{\partial r}{\partial a} &= \sum_{j=Q}^R [2a - 2p_{yj} + 2b \cdot p_{xj} + 2c \cdot p_{zj}] = 0 \\ \Rightarrow a &= \frac{\sum_{j=Q}^R [p_{yj} - b \cdot p_{xj} - c \cdot p_{zj}]}{n}, \quad \text{mit } n = R - Q + 1 \quad . \end{aligned} \quad (\text{A.4})$$

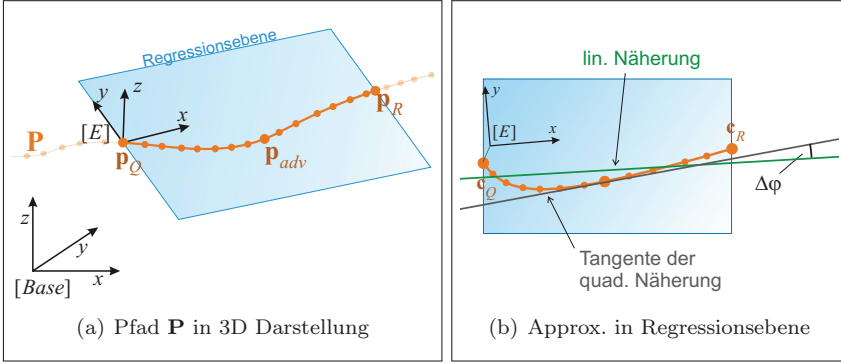


Abbildung A.1: Berechnung einer Tangente durch lineare/quadratische Pfadapproximation in \mathbb{R}^3 nach Projektion des Pfades P auf eine Regressionsebene.

Aus $\frac{\partial r}{\partial b} = 0$ und $\frac{\partial r}{\partial c} = 0$ ergeben sich entsprechend:

$$b = \frac{n \cdot \overline{p_x} \cdot \overline{p_y} - s_{xy} - n \cdot \overline{p_x} \cdot c \cdot \overline{p_z} + c \cdot s_{xz}}{n \cdot \overline{p_x}^2 - s_{xx}} \quad (\text{A.5})$$

$$c = \frac{n \cdot s_{yz} \cdot \overline{p_x}^2 - \overline{p_x} \cdot n \cdot \overline{p_y} \cdot s_{xz} - \overline{p_x} \cdot n \cdot s_{xy} \cdot \overline{p_z} + s_{xx} \cdot n \cdot \overline{p_y} \cdot \overline{p_z} + s_{xy} \cdot s_{xz} - s_{xx} \cdot s_{yz}}{n \cdot s_{zz} \cdot \overline{p_x}^2 - 2 \cdot n \cdot s_{xz} \cdot \overline{p_z} \cdot \overline{p_x} + s_{xx} \cdot n \cdot \overline{p_z}^2 + s_{zz}^2 - s_{xx} \cdot s_{zz}} \quad (\text{A.6})$$

mit den Abkürzungen

$$\begin{aligned} s_{xx} &= \sum_{j=1}^n p_{xj}^2; & s_{xy} &= \sum_{j=1}^n p_{xj} p_{yj}; & s_{xz} &= \sum_{j=1}^n p_{xj} p_{zj}; \\ s_{yz} &= \sum_{j=1}^n p_{yj} p_{zj}; & s_{zz} &= \sum_{j=1}^n p_{zj}^2; & \bar{p} &= \frac{1}{n} \sum_{j=1}^n p_j \end{aligned} \quad (\text{A.7})$$

Die Ebenengleichung A.2 der Regressionsebene lässt sich mit dem Normalenvektor \vec{n}_R in Normalendarstellung folgendermaßen beschreiben:

$$\begin{aligned} \vec{E}_R: \quad 1 &= -\frac{b}{a} \cdot x + \frac{1}{a} \cdot y - \frac{c}{a} \cdot z \quad \Rightarrow \quad \vec{n}_R \cdot (\mathbf{x} - \mathbf{e}_0) = 0 \\ \text{mit } \vec{n}_R &= \left(-\frac{b}{a}, \frac{1}{a}, -\frac{c}{a}\right)^T; \quad \mathbf{e}_0 = (0, a, 0)^T \end{aligned} \quad (\text{A.8})$$

Für die Regression muss jeder Messwert \mathbf{p}_j auf die Ebene projiziert werden, indem der Schnittpunkt \mathbf{c}_j zwischen der Ebene \vec{E}_R und der Geraden \vec{g}_j durch \mathbf{p}_j in Richtung des Normalenvektors \vec{n}_R berechnet wird:

$$\vec{g}_j : \mathbf{x} = \mathbf{p}_j + \lambda_j \cdot \vec{n}_R \quad (\text{A.9})$$

$$\vec{E}_R = \vec{g}_j \Rightarrow \lambda_j = \frac{\vec{n}_R \cdot (\mathbf{e}_0 - \mathbf{p}_j)}{\vec{n}_R \cdot \vec{n}_R} ; \quad \mathbf{c}_j = \mathbf{p}_j + \lambda_j \cdot \vec{n}_R \quad . \quad (\text{A.10})$$

Die Punkte \mathbf{c}_j liegen in der Ebene \vec{E}_R . Das zu dieser Ebene gehörige Koordinatensystem $K_{[E]}$ wird im Basissystem folgendermaßen definiert (vgl. Abbildung A.1):

$$\begin{aligned} \vec{a}_E &= \frac{\vec{n}_R}{\|\vec{n}_R\|}; & \vec{n}_E &= \frac{\mathbf{c}_n - \mathbf{c}_1}{\|\mathbf{c}_n - \mathbf{c}_1\|}; & \vec{o}_E &= \vec{a}_E \times \vec{n}_E \\ \mathbf{T}_{base}^E &= \begin{pmatrix} \vec{n}_E & \vec{o}_E & \vec{a}_E & \mathbf{c}_1 \\ 0 & 0 & 0 & 1 \end{pmatrix} . \end{aligned} \quad (\text{A.11})$$

Um die Punkte \mathbf{c}_j in \mathbb{R}^2 zu beschreiben, wird jeder Punkt in das Ebenenkoordinatensystem $K_{[E]}$ transformiert:

$$\mathbf{p}_j^E = (\mathbf{T}_{base}^E)^{-1} \cdot \begin{pmatrix} \mathbf{c}_j \\ 1 \end{pmatrix} \quad (\text{A.12})$$

Da die z -Achse des Ebenenkoordinatensystems durch den Normalenvektor von \vec{E}_R definiert ist, ist die z -Komponente der in die Ebene projizierten Punkte $\mathbf{p}_{zj}^E = 0$. Die Regressionsgerade mit den Parametern a_g und b_g in \mathbb{R}^2 kann daraufhin mit den x - und y -Komponenten von \mathbf{p}_j^E berechnet werden:

$$y = a_g + b_g \cdot x \quad \text{mit} \quad b_g = \frac{\sum_{j=1}^n \left(p_{xj}^E - \overline{p_x^E} \right) \left(p_{yj}^E - \overline{p_y^E} \right)}{\sum_{j=1}^n \left(p_{xj}^E - \overline{p_x^E} \right)^2}; \quad a_g = \overline{p_y^E} - b_g \cdot \overline{p_x^E} . \quad (\text{A.13})$$

Anschließend werden zwei Punkte der Regressionsgerade mittels der Transformation \mathbf{T}_{base}^E in den dreidimensionalen Raum zurück transformiert, um die Tangente in \mathbb{R}^3 zu beschreiben.

B Quadratische Regression

Analog zu der linearen Regression in Anhang A wird die quadratische Regression in der Projektion auf einer Regressionsebene berechnet. Im Folgenden wird die Berechnung der quadratischen Regression in \mathbb{R}^2 dargestellt. Die quadratische Funktion k wird in Abhängigkeit der Koordinaten x und y folgendermaßen beschrieben:

$$k: y = a_k + b_k \cdot x + c_k \cdot x^2 \quad . \quad (\text{B.1})$$

Die unbekannten Parameter a_k, b_k, c_k werden mithilfe der minimalen Fehlerquadratmethode aus den Messwerten $x_i, y_i, (i = 1, 2, \dots, n)$ ermittelt. Daraus ergibt sich die folgende, zu minimierende Forderung r :

$$r = \sum_{i=1}^n [y_i - (a + b \cdot x_i + c \cdot x_i^2)]^2 \rightarrow \min \quad (\text{B.2})$$

Nullsetzen der partiellen Ableitung liefert für a :

$$\frac{\partial r}{\partial a} = \sum_{i=1}^n [2a - 2y_i + 2bx_i + 2cx_i^2] = 0 \quad \Rightarrow a = \frac{\sum_{i=1}^n [y_i - bx_i - cx_i^2]}{n} \quad (\text{B.3})$$

Für die Parameter b und c ergibt sich entsprechend (Berechnung mit Maple, Index i für die Lesbarkeit weggelassen):

$$c = \frac{\overline{yx^2} \cdot (\overline{x})^2 - \overline{x} \cdot \overline{x^3} \cdot \overline{y} - \overline{x} \cdot \overline{x^2} \cdot \overline{yx} + \overline{yx} \cdot \overline{x^3} - \overline{x^2} \cdot \overline{yx^2} + \left(\overline{x^2}\right)^2 \cdot \overline{y}}{\overline{x^4} \cdot (\overline{x})^2 - 2 \cdot \overline{x^2} \cdot \overline{x} \cdot \overline{x^3} + \left(\overline{x^2}\right)^3 + \left(\overline{x^3}\right)^2 - \overline{x^2} \cdot \overline{x^4}} \quad (\text{B.4})$$

$$b = \frac{-n \cdot \overline{x} \cdot \overline{y} + n \cdot \overline{yx} + n \cdot \overline{x} \cdot c \cdot \overline{x^2} - c \cdot n \cdot \overline{x^3}}{-n \cdot (\overline{x})^2 + n \cdot \overline{x^2}} \quad (\text{B.5})$$

Der Überstrich kennzeichnet den Mittelwert des jeweiligen Messwertes zwischen $1, \dots, n$.

C Identifikation

Dieser Abschnitt beschreibt die Identifikation nach der Methode der minimalen Fehlerquadrate [Ise92]. Es wird von dem folgenden, nicht sprungfähigen, stabilen und zeitinvarianten System zweiter Ordnung mit der Totzeit d ausgegangen. Das diskrete System und die Differenzengleichung des Systems lauten wie folgt:

$$G(z) = \frac{y(z)}{u(z)} = \frac{B(z^{-1})}{A(z^{-1})} z^{-d} = \frac{b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}} z^{-d} \quad (\text{C.1})$$

$$\begin{aligned} \Rightarrow y(k) + a_1 y(k-1) + a_2 y(k-2) = \\ b_1 u(k-d-1) + b_2 u(k-d-2) \end{aligned} \quad (\text{C.2})$$

mit

$$u(k) = U(k) - U_{00}, \quad y(k) = Y(k) - Y_{00} \quad . \quad (\text{C.3})$$

U_{00} , Y_{00} sind exakt messbare und zueinander gehörige Beharrungswerte von $u(k)$, $y(k)$. $U(k)$, $Y(k)$ sind die messbaren Änderungen von $u(k)$, $y(k)$.

Mit den Parameterschätzwerten und Gleichung C.2 wird folgende Einschrittvorhersage gemacht (die Vorhersage beruht darauf, dass zum aktuellen Schritt k nur Messwerte und Parameter von $(k-1)$ vorhanden sind):

$$\begin{aligned} \hat{y}(k|k-1) &= -\hat{a}_1(k-1)y(k-1) - \hat{a}_2(k-1)y(k-2) \\ &\quad + \hat{b}_1(k-1)u(k-d-1) + \hat{b}_2(k-1)u(k-d-2) \quad (\text{C.4}) \\ &= \Psi^T(k) \hat{\theta}(k-1) \quad , \end{aligned}$$

mit dem Datenvektor Ψ^T und dem Parametervektor $\hat{\theta}$:

$$\begin{aligned} \Psi^T(k) &= [-y(k-1), -y(k-2), u(k-d-1), u(k-d-2)] \\ \hat{\theta} &= [\hat{a}_1, \hat{a}_2, \hat{b}_1, \hat{b}_2]^T \quad . \end{aligned} \quad (\text{C.5})$$

Mit der geschätzten Vorhersage des Ausgangs \hat{y} zum Zeitpunkt k und der Messung y zum Zeitpunkt k gilt für den Gleichungsfehler

$$\begin{aligned} e(k) &= y(k) - \hat{y}(k|k-1) = y(k) - \Psi^T(k)\hat{\theta}(k-1) \\ \Rightarrow y(k) &= \Psi^T(k)\hat{\theta}(k-1) + e(k) \quad . \end{aligned} \quad (\text{C.6})$$

Aus N Messungen ergibt sich ein Gleichungssystem mit N Gleichungen. Zur Unterdrückung von Störsignalen werden sehr viel mehr Messungen verwendet, als zur Lösung des Gleichungssystems notwendig wäre. Das Gütekriterium V ergibt sich zu:

$$V = \sum_{k=m+d}^{k=m+d+N} e^2(k) \quad . \quad (\text{C.7})$$

Mit der Ordnung m des Systems und der Anzahl der Messungen N wird V nach $\hat{\theta}$ folgendermaßen minimiert:

$$\mathbf{\Psi} = [\Psi(m), \Psi(m+1), \dots, \Psi(N)]^T \quad (\text{C.8})$$

$$\mathbf{y} = [y(m), y(m+1), \dots, y(N)]^T \quad (\text{C.9})$$

$$\frac{dV}{d\hat{\theta}} = -2\mathbf{\Psi}^T(\mathbf{y} - \mathbf{\Psi}^T\hat{\theta}) = 0 \quad (\text{C.10})$$

$$\hat{\theta} = \left[\mathbf{\Psi}^T \mathbf{\Psi} \right]^{-1} \mathbf{\Psi}^T \mathbf{y} \quad . \quad (\text{C.11})$$

Gleichung C.11 liefert die Schätzung der Parameter eines Feder-Masse-Dämpfer Systems aus N Messungen. Nach Gleichung C.3 ist jedoch die Kenntnis eines Ruhezustandes notwendig. Da die Identifikation während der Bewegung entlang einer unbekannt geformten Kontur erfolgen soll, kann einer Kontaktkraft die entsprechende Verformung der Umgebung nicht direkt zugeordnet werden. Jedoch kann die Identifikation erweitert werden, sodass die Identifikation der Ruheposition zu den unbekannten Parametern gehört. Der Ruheposition Y_{00} der Feder mit der Steifigkeit

K wird die Kontaktkraft $U_{00} = 0N$ zugeordnet. Daraus ergeben sich die erweiterten Systemgleichungen wie folgt:

$$G(z) = \frac{Y(z)}{U(z) + K \cdot Y_{00}} = \frac{Y(z)}{U(z) + U^*} \quad (\text{C.12})$$

$$\begin{aligned} y(k) + a_1 y(k-1) + a_2 y(k-2) = \\ b_1 (u(k-d-1) + u^*) + b_2 (u(k-d-2) + u^*) \end{aligned} \quad (\text{C.13})$$

$$\Psi^T(k) = [-Y_{k-1}, -Y_{k-2}, U_{k-d-1}, U_{k-d-2}, 1] \quad (\text{C.14})$$

$$\hat{\theta} = [\hat{a}_1, \hat{a}_2, \hat{b}_1, \hat{b}_2, \hat{c}]^T \quad \text{mit } \hat{c} = (b_1 + b_2) \cdot K \cdot Y_{00} \quad . \quad (\text{C.15})$$

D Bilineare Transformation

Die Bilineare Transformation (auch Tustin Approximation genannt), wird zur Transformation einer Gleichung vom zeitdiskreten in den kontinuierlichen Raum verwendet [OSB04]. Das folgende zeitdiskrete Modell soll in den kontinuierlichen Raum transformiert werden:

$$G(z) = \frac{\hat{b}_1 \cdot z^{-1} + \hat{b}_2 \cdot z^{-2}}{1 + \hat{a}_1 \cdot z^{-1} + \hat{a}_2 \cdot z^{-2}} = \frac{\hat{b}_1 \cdot z + \hat{b}_2}{z^2 + \hat{a}_1 \cdot z + \hat{a}_2} \quad (\text{D.1})$$

Mittels der Tustin Näherung wird z folgendermaßen ersetzt:

$$z = \frac{1 + \frac{1}{2} \cdot s \cdot T}{1 - \frac{1}{2} \cdot s \cdot T} \quad (\text{D.2})$$

Durch Einsetzen von Gleichung D.2 in Gleichung D.1 ergibt sich die folgende (genäherte) Übertragungsfunktion im Frequenzbereich:

$$G(s) = \frac{b_{s,2} \cdot s^2 + b_{s,1} \cdot s + b_{s,0}}{a_{s,2} \cdot s^2 + a_{s,1} \cdot s + a_{s,0}} \quad (\text{D.3})$$

Durch Einsetzen der Zahlenwerte aus Experimenten in $G(z)$ aus der Identifikation wird deutlich, dass $b_{s,2}$ und $b_{s,1}$ im Verhältnis zu $b_{s,0}$ einen sehr geringen Anteil haben – werden daher als Null angenommen. Daraus ergibt sich folgende Form für $G(s)$. Diese Form lässt sich mit dem Modell des Feder-Masse-Dämpfersystems gleichsetzen:

$$G(s) = \frac{b_{s,0}}{a_{s,2} \cdot s^2 + a_{s,1} \cdot s + a_{s,0}} = \frac{1}{M \cdot s^2 + D \cdot s + K} = \frac{\frac{1}{M}}{s^2 + \frac{D}{M} \cdot s + \frac{K}{M}} \quad (\text{D.4})$$

mit

$$M = \frac{a_{s,2}}{b_{s,0}}; \quad D = \frac{a_{s,1}}{a_{s,2}} \cdot M; \quad K = \frac{a_{s,0}}{a_{s,2}} \cdot M \quad (\text{D.5})$$

$$(\text{D.6})$$

Abhängig von Gleichung D.1 ergeben sich die folgenden Zusammenhänge der Parameter:

$$\hat{M} = -\frac{1}{4} \cdot \frac{T^2 \cdot (\hat{a}_1 - 1 - \hat{a}_2)}{\hat{b}_1 + \hat{b}_2} \quad (\text{D.7})$$

$$\hat{D} = -\frac{T \cdot (\hat{a}_2 - 1)}{\hat{b}_1 + \hat{b}_2} \quad (\text{D.8})$$

$$\hat{K} = \frac{1 + \hat{a}_2 + \hat{a}_1}{\hat{b}_1 + \hat{b}_2} \quad (\text{D.9})$$

Publikationen

- WEBER, W. ; KOCH, H.: A State-Space Controller for Movement Axes with Flexibilities. In: *Automatisierungstechnische Praxis atp edition, No.4/12, S. 20–24, 2010.*
- KOCH, H. ; KÖNIG, A. ; WEIGL-SEITZ, A. ; KLEINMANN, K.: Improving Force Measurements in Multimodal Robotic Contour Following Tasks Using Acceleration Sensors. In: *Proceedings of the 12th Mechatronics Forum Biennial International Conference, Zurich, Switzerland, 2010, S. 65–72*
- KOCH, H. ; KÖNIG, A. ; KLEINMANN, K. ; WEIGL-SEITZ, A. ; SUCHÝ, J.: Predictive Robotic Contour Following Using Laser Camera Triangulation. In: *International Conference on Advanced Intelligent Mechatronics, Budapest, Hungary, 2011, S. 422–427*
- KOCH, H. ; KÖNIG, A. ; KLEINMANN, K. ; WEIGL-SEITZ, A. ; SUCHÝ, J.: Force, Acceleration and Vision Sensor Fusion for Contour Following Tasks with an Industrial Robot. In: *IEEE International Conference on Robotic and Sensors Environments (ROSE), Montreal, Canada, 2011, S. 1–6*
- KOCH, H. ; KÖNIG, A. ; KLEINMANN, K. ; WEIGL-SEITZ, A. ; SUCHÝ, J.: Multisensor Robotic Contour Following on Deformable Objects. In: *IASTED International Conference on Robotics, Pittsburgh, USA, 2011, S. 150–156*
- KOCH, H. ; SUCHÝ, J.: Multisensor-Konturverfolgung an nachgiebigen Objekten. Vortrag: 46. *Regelungstechnisches Kolloquium, Boppard, Germany, 2012*

- KOCH, H. ; KÖNIG, A. ; KLEINMANN, K. ; WEIGL-SEITZ, A. ; SUCHÝ, J.: Filtering and Corner Detection in Predictive Robotic Contour Following. In: *7th German Conference on Robotics, Munich, Germany*, 2012, S. 502–507
- KOCH, H. ; KÖNIG, A. ; KLEINMANN, K. ; WEIGL-SEITZ, A. ; SUCHÝ, J.: Multisensor Contour Following with Vision, Force and Acceleration Sensors for an Industrial Robot. In: *IEEE Transactions on Instrumentation & Measurement* Vol. 62, Nr. 2, Februar 2013, S. 268–280,

Literaturverzeichnis

- [ADTS07] AMARASINGHE, Ranjith ; DAO, Dzung V. ; TORIYAMA, Toshioyuki ; SUGIYAMA, Susumu: Development of miniaturized 6-axis accelerometer utilizing piezoresistive sensing elements. In: *Sensors and Actuators A: Physical* 134 (2007), Nr. 2, S. 310–320
- [Agi85] AGIN, G.: Calibration and use of a light stripe range sensor mounted on the hand of a robot. In: *Proceedings of IEEE International Conference on Robotics and Automation* Bd. 2, 1985, S. 680–685
- [AH84] ÅSTRÖM, K.J. ; HÄGGLUND, T.: Automatic tuning of simple regulators with specifications on phase and amplitude margins. In: *Automatica* 20 (1984), Nr. 5, S. 645–651
- [ATH99] ASADA, M. ; TANAKA, T. ; HOSODA, K.: Visual tracking of unknown moving object by adaptive binocular visual servoing. In: *Proceedings of IEEE/SICE/RSJ International Conference on Multisensor Fusion and Integration for Intelligent Systems*, 1999, S. 249–254
- [BDS01] BAETEN, J. ; DE SCHUTTER, J.: Combined vision/force control at corners in planar robotic contour following. In: *Proceedings of IEEE/ASME International Conference on Advanced Intelligent Mechatronics* Bd. 2, 2001, S. 810–815
- [BDSB03] BAETEN, J. ; DE SCHUTTER, J. ; BRUYNINCKX, H.: Integrated Vision/Force Robotic Servoing in the Task Frame Formalism. In: *The International Journal of Robotics Research* Bd. 22, 2003, S. 941–954
- [BRB⁺06] BRYZEK, J. ; ROUNDY, S. ; BIRCUMSHAW, B. ; CHUNG, C. ; CASTELLINO, K. ; STETTER, J.R. ; VESTEL, M.: Marvelous

- MEMS. In: *IEEE Circuits and Devices Magazine* 22 (2006), Mär, Nr. 2, S. 8–28
- [BS99] BAETEN, Johan ; SCHUTTER, Joris D.: Improving force controlled planar contour following using online eye-in-hand vision based feedforward. In: *Proceedings of IEEE/ASME International Conference on Advanced Intelligent Mechatronics* (1999), Jul, S. 902–907
- [BS02] BAETEN, J. ; SCHUTTER, J. D.: Hybrid vision/force control at corners in planar robotic-contour following. In: *IEEE/ASME Transactions on Mechatronics* 7 (2002), Jun, Nr. 2, S. 143–151
- [BS04] BAETEN, Johan ; SCHUTTER, Joris D.: *Integrated Visual Servoing and Force Control*. Springer, 2004 (Springer Tracts in Advanced Robotics 8)
- [BSMM01] BRONSTEIN ; SEMENDJAJEW ; MUSIOL ; MÜHLIG: *Taschenbuch der Mathematik*. 5. Auflage. Verlag Harri Deutsch, 2001
- [CC00] COLLEWET, C. ; CHAUMETTE, F.: A contour approach for image-based control on objects with complex shape. In: *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems* Bd. 1, 2000, S. 751–756
- [CG96] CORKE, P.I. ; GOOD, M.C.: Dynamic effects in visual closed-loop systems. In: *IEEE Transactions on Robotics and Automation* 12 (1996), Okt, Nr. 5, S. 671–683
- [CH01] CORKE, P.I. ; HUTCHINSON, S.A.: A new partitioned approach to image-based visual servo control. In: *IEEE Transactions on Robotics and Automation* 17 (2001), Aug, Nr. 4, S. 507–515
- [CH02] CHESI, G. ; HASHIMOTO, K.: Static-eye against hand-eye visual servoing. In: *Proceedings of the 41st IEEE Conference on Decision and Control* Bd. 3, 2002, S. 2854–2859
- [CH06] CHAUMETTE, Francois ; HUTCHINSON, Seth: Visual Servo Control Part I: Basic Approaches. In: *IEEE Robotics & Automation Magazine* 13 (2006), Dez, Nr. 4, S. 82–90

- [CL07] CHENG, Ming-Yang ; LEE, Cheng-Chien: Motion Controller Design for Contour-Following Tasks Based on Real-Time Contour Error Estimation. In: *IEEE Transactions on Industrial Electronics* 54 (2007), Jun, Nr. 3, S. 1686–1695
- [CMCK93] COSTE-MANIÈRE, Eve ; COUVIGNOU, Philippe ; KHOSLA, Pradeep K.: Robotic Contour Following based on Visual Servoing. In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems* (1993), S. 716–722
- [CMCK95] COSTE-MANIÈRE, Eve ; COUVIGNOU, P. ; KHOSLA, P.: Visual Servoing in the Task-Function Framework: a Contour Following Task. In: *Journal of Intelligent and Robotic Systems* 12 (1995), S. 1–21
- [CMT87] CLARKE, D ; MOHTADI, C ; TUFFS, P: Generalized predictive control. In: *Automatica* 23 (1987), Mär, Nr. 2, S. 137–148
- [Cor94] CORKE, Peter I.: Visual Control Of Robot Manipulators – A Review. In: *Visual Servoing*, World Scientific, 1994, S. 1–31
- [Cor95] CORKE, Peter I.: Dynamic Issues in Robot Visual-Servo Systems. In: *Int. Symp. on Robotics Research*, Springer, 1995, S. 488–498
- [CR79] CRAIG, J.J. ; RAIBERT, M.H.: A systematic method of hybrid position/force control of a manipulator. In: *Proceedings of IEEE International Computer Software and Applications Conference*, 1979, S. 446–451
- [Deg98] DEGUCHI, K.: Optimal motion control for image-based visual servoing by decoupling translation and rotation. In: *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems* Bd. 2, 1998, S. 705–711
- [DJSW02] DENG, Lingfeng ; JANABI-SHARIFI, F. ; WILSON, W.J.: Stability and robustness of visual servoing methods. In: *Proceedings of IEEE International Conference on Robotics and Automation* Bd. 2, 2002, S. 1604–1609
- [DS95] DEMEY, Sabine ; SCHUTTER, Joris D.: Fast and high-quality planar contour following in the presence of large position

- uncertainties. In: *Proceedings of IEEE International Conference on Robotics and Automation 2* (1995), S. 2096–2101
- [DSB88] DE SCHUTTER, J. ; BRUSSEL, H. van: Compliant Robot Motion I. A Formalism for Specifying Compliant Motion Tasks. In: *The International Journal of Robotics Research* Bd. 7, 1988, S. 3–17
- [ECR92] ESPIAU, B. ; CHAUMETTE, F. ; RIVES, P.: A new approach to visual servoing in robotics. In: *IEEE Transactions on Robotics and Automation* 8 (1992), Jun, Nr. 3, S. 313–326
- [EL00] ELLIS, G. ; LORENZ, R.D.: Resonant load control methods for industrial servo drives. In: *Conference Record of the 2000 IEEE Industry Applications Conference* 3 (2000), S. 1438–1445
- [Esp94] ESPIAU, Bernard: Effect of camera calibration errors on visual servoing in robotics. In: YOSHIKAWA, Tsuneo (Hrsg.) ; MIYAZAKI, Fumio (Hrsg.): *Experimental Robotics III* Bd. 200. Springer Berlin / Heidelberg, 1994, S. 182–192
- [FCM00] FLANDIN, G. ; CHAUMETTE, F. ; MARCHAND, E.: Eye-in-hand/eye-to-hand cooperation for visual servoing. In: *Proceedings of IEEE International Conference on Robotics and Automation* Bd. 3, 2000, S. 2741–2746
- [GAMJ05] GRAAF, M.W. de ; AARTS, R.G.K.M. ; MEIJER, J. ; JONKER, J.B.: Robot-sensor synchronization for real-time seam-tracking in robotic laser welding. In: *Proceedings of the Third International WLT-Conference on Lasers in Manufacturing, Munich* (2005), Jun
- [GM02] GANGLOFF, J.A. ; MATHELIN, M.F. de: Visual servoing of a 6-DOF manipulator for unknown 3-d profile following. In: *IEEE Transactions on Robotics and Automation* 18 (2002), Aug, Nr. 4, S. 511–520
- [GROJ04] GARCIA, J.G. ; ROBERTSSON, A. ; ORTEGA, J.G. ; JOHANSSON, R.: Sensor fusion of force and acceleration for robot force control. In: *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems* Bd. 3, 2004, S. 3009–3014

- [GROJ05] GARCIA, J.G. ; ROBERTSSON, A. ; ORTEGA, J.G. ; JOHANSSON, R.: Force and Acceleration Sensor Fusion for Compliant Robot Motion Control. In: *Proceedings of the IEEE International Conference on Robotics and Automation*, 2005, S. 2709–2714
- [GTK10] GHAFFARI TOISERKAN, K. ; KÖVECSES, J.: Improving stability and performance of digitally controlled systems: The concept of Modified Holds. In: *IEEE International Conference on Robotics and Automation*, 2010, S. 5173–5180
- [Has03] HASHIMOTO, Koichi: A review on vision-based control of robot manipulators. In: *Advanced Robotics* 17 (2003), Nr. 10, S. 969–991
- [HHC96] HUTCHINSON, S. ; HAGER, G.D. ; CORKE, P.I.: A tutorial on visual servo control. In: *IEEE Transactions on Robotics and Automation* 12 (1996), Okt, Nr. 5, S. 651–670
- [HKEK91] HASHIMOTO, K. ; KIMOTO, T. ; EBINE, T. ; KIMURA, H.: Manipulator control with image-based visual servo. In: *Proceedings of IEEE International Conference on Robotics and Automation* Bd. 3, 1991, S. 2267–2271
- [Hog85] HOGAN, Neville: Impedance Control: An Approach to Manipulation: Part I—Theory. In: *Journal of Dynamic Systems, Measurement, and Control* 107 (1985), Nr. 1
- [Ise92] ISERMANN, Rolf: *Identifikation dynamischer Systeme 1. 2. Auflage*. Springer-Verlag, 1992
- [JABH07] JEON, Sang-Wook ; AHN, Doo-Sung ; BAE, Hyo-Jeong ; HONG, Chang-Woo: Object contour following task based on integrated information of vision and force sensor. In: *International Conference on Control, Automation and Systems* (2007), Okt, S. 1040–1045
- [JSDW11] JANABI-SHARIFI, F. ; DENG, Lingfeng ; WILSON, W.J.: Comparison of Basic Visual Servoing Methods. In: *IEEE/ASME Transactions on Mechatronics* 16 (2011), Okt, Nr. 5, S. 967–983

- [KES01] KUNZ, K. ; ENOKSSON, P. ; STEMME, G.: Highly sensitive triaxial silicon accelerometer with integrated PZT thin film detectors. In: *Sensors and Actuators A: Physical*, 2001, S. 156–160
- [KKK⁺11a] KOCH, H. ; KÖNIG, A. ; KLEINMANN, K. ; WEIGL-SEITZ, A. ; SUCHÝ, J.: Force, Acceleration and Vision Sensor Fusion for Contour Following Tasks with an Industrial Robot. In: *IEEE International Conference on Robotic and Sensors Environments (ROSE)*, Montreal, Canada, 2011, S. 1–6
- [KKK⁺11b] KOCH, H. ; KÖNIG, A. ; KLEINMANN, K. ; WEIGL-SEITZ, A. ; SUCHÝ, J.: Multisensor Robotic Contour Following on Deformable Objects. In: *IASTED International Conference on Robotics*, Pittsburgh, USA, 2011, S. 150–156
- [KKK⁺11c] KOCH, H. ; KÖNIG, A. ; KLEINMANN, K. ; WEIGL-SEITZ, A. ; SUCHÝ, J.: Predictive Robotic Contour Following Using Laser Camera Triangulation. In: *International Conference on Advanced Intelligent Mechatronics*, Budapest, Hungary, 2011, S. 422–427
- [KKK⁺12] KOCH, H. ; KÖNIG, A. ; KLEINMANN, K. ; WEIGL-SEITZ, A. ; SUCHÝ, J.: Filtering and Corner Detection in Predictive Robotic Contour Following. In: *7th German Conference on Robotics*, Munich, Germany, 2012, S. 502–507
- [KKW08] KRÖGER, T. ; KUBUS, D. ; WAHL, F.M.: 12D force and acceleration sensing: A helpful experience report on sensor characteristics. In: *IEEE International Conference on Robotics and Automation*, 2008, S. 3455–3462
- [KKWSK10] KOCH, H. ; KÖNIG, A. ; WEIGL-SEITZ, A. ; KLEINMANN, K.: Improving Force Measurements in Multimodal Robotic Contour Following Tasks Using Acceleration Sensors. In: *Proceedings of the 12th Mechatronics Forum Biennial International Conference*, Zurich, Switzerland (2010)
- [KLY96] KIM, Whee-Kuk ; LEE, Jun-Yong ; YI, Byung-Ju: RCC characteristics of planar/spherical three degree of freedom parallel mechanisms with joint compliances. In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems* Bd. 1, 1996, S. 360–367

- [KMN⁺93] KASE, H. ; MARU, N. ; NISHIKAWA, A. ; YAMADA, S. ; MIYAZAKI, F.: Visual servoing of the manipulator using the stereo vision. In: *Proceedings of International Conference on Industrial Electronics, Control, and Instrumentation*, 1993
- [KSCK95] KIM, Jae S. ; SON, Young T. ; CHO, Hyung S. ; KOH, Kwang I.: A robust method for vision-based seam tracking in robotic arcwelding. In: *Proceedings of the IEEE International Symposium on Intelligent Control* (1995), S. 363–368
- [LC04] LUO, Hong ; CHEN, Xiaoqi: Laser visual sensing for seam tracking in robotic are welding of titanium alloys. In: *International Journal of Advanced Manufacturing Technology* (2004), Mai, S. 1012–1017
- [Lee05] LEE, Sangcheol: Development of a new variable remote center compliance (VRCC) with modified elastomer shear pad (ESP) for robot assembly. In: *IEEE Transactions on Automation Science and Engineering* 2 (2005), Apr, Nr. 2, S. 193–197
- [LN00] LUO, Yanhong ; NELSON, Bradley J.: Fusing force and vision feedback for manipulating deformable objects. In: *Journal of Robotic Systems* Bd. 3, 2000, S. 103–117
- [LW07] LUTZ, Holger ; WENDT, Wolfgang: *Taschenbuch der Regelungstechnik*. 7. Auflage. Verlag Harri Deutsch, 2007
- [LWH98] LANGE, F. ; WUNSCH, P. ; HIRZINGER, G.: Predictive vision based control of high speed industrial robot paths. In: *Proceedings of IEEE International Conference on Robotics and Automation* Bd. 3, 1998, S. 2646–2651
- [LWWL06] LIU, Yun-Hui ; WANG, Hesheng ; WANG, Chengyou ; LAM, Kin K.: Uncalibrated visual servoing of robots using a depth-independent interaction matrix. In: *IEEE Transactions on Robotics* 22 (2006), Aug, Nr. 4, S. 804–817
- [Mas79] MASON, Matthew T.: *Compliance and Force Control for Computer Controlled Manipulators*, Masterthesis, Apr 1979
- [Mas81] MASON, Matthew T.: Compliance and Force Control for Computer Controlled Manipulators. In: *IEEE Transactions*

- on Systems, Man and Cybernetics* 11 (1981), Jun, Nr. 6, S. 418–432
- [MM06] MASAOKI, K. ; MINORU, Y.: Development of a Vision Based 3D Range Sensor using Modulated Light. In: *International Joint Conference SICE-ICASE*, 2006, S. 2300–2305
- [MMB98] MOREL, G. ; MALIS, E. ; BOUDET, S.: Impedance based combination of visual and force control. In: *Proceedings of IEEE International Conference on Robotics and Automation* Bd. 2, 1998, S. 1743–1748
- [MS98] MOKHTARIAN, F. ; SUOMELA, R.: Robust image corner detection through curvature scale space. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 20 (1998), Nr. 12, S. 1376–1381
- [Nev76] NEVATIA, Ramakant: Depth measurement by motion stereo. In: *Computer Graphics and Image Processing* 5 (1976), Nr. 2, S. 203–214
- [NHNT00] NAGAHAMA, K. ; HASHIMOTO, K. ; NORITSUGU, T. ; TAKAIKAWA, M.: Visual servoing based on object motion estimation. In: *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems* Bd. 1, 2000, S. 245–250
- [NMK95] NELSON, Bradley J. ; MORROW, J. D. ; KHOSLA, Pradeep K.: Improved force control through visual servoing. In: *American Control Conference*, 1995, S. 380–386
- [NMK96] NELSON, B. J. ; MORROW, J. D. ; KHOSLA, P. K.: Robotic manipulation using high bandwidth force and vision feedback. In: *Mathematical and Computer Modelling* 24 (1996), S. 11–29
- [Ols10] OLSON, Edwin: A passive solution to the sensor synchronization problem. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems* (2010), S. 1059–1064
- [OSB04] OPPENHEIM, Alan V. ; SCHAFER, Ronald W. ; BUCK, John R.: *Zeitdiskrete Signalverarbeitung*. Oldenbourg Verlag, 2004

- [Ott10] OTTO, Marc: *Konzeption und Realisierung eines robotergeführten 3D Lasertriangulationssensors mit einer intelligenten Kamera*, Hochschule Darmstadt - University of Applied Sciences, Diplomarbeit, 2010
- [PR98] PUERS, R. ; REYNTJENS, S.: Design and processing experiments of a new miniaturized capacitive triaxial accelerometer. In: *Sensors and Actuators A: Physical* 68 (1998), S. 324–328
- [PSAR09] PARI, L. ; SEBASTIAN, J.M. ; ANGEL, L. ; RUEDA, J.D.: Image Base Visual Servoing: Estimation of the image Jacobian by using lines in a stereo vision system. In: *IEEE International Conference on E-Learning in Industrial Electronics*, 2009, S. 109–114
- [RC81] RAIBERT, M. H. ; CRAIG, J. J.: Hybrid Position/Force Control of Manipulators. In: *Journal of Dynamic Systems, Measurement, and Control* 103 (1981), Nr. 2, S. 126–133
- [Sal80] SALISBURY, J. K.: Active stiffness control of a manipulator in cartesian coordinates. In: *19th IEEE Conference on Decision and Control including the Symposium on Adaptive Processes* Bd. 19, 1980, S. 95–100
- [SHL02] SIM, T.P. ; HONG, G.S. ; LIM, K.B.: A pragmatic 3D visual servoing system. In: *Proceedings of IEEE International Conference on Robotics and Automation* Bd. 4, 2002, S. 4185–4190
- [SLH⁺09] SUYI, Liu ; LINGTENG, Liu ; HUA, Zhang ; JIANJUN, Bai ; GUORONG, Wang: Study of robot seam tracking system with laser vision. In: *International Conference on Mechatronics and Automation*, 2009, S. 1296–1301
- [Smi59] SMITH, O.J.M.: A controller to overcome dead-time. In: *ISA Journal* 6 (1959), Nr. 2, S. 28–33
- [SSVO09] SICILIANO, Bruno ; SCIAVICCO, Lorenzo ; VILLANI, Luigi ; ORIOLO, Guiseppe: *Robotics Modelling, Planning and Control*. 2. Auflage. Springer, 2009

- [Tsa87] TSAI, R.: A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses. In: *IEEE Journal of Robotics and Automation* 3 (1987), Aug, Nr. 4, S. 323–344
- [Unb07] UNBEHAUEN, H.: *Regelungstechnik I*. 14. Auflage. Vieweg-Verlag, 2007
- [VK92] VOLPE, Richard ; KHOSLA, Pradeep: An Experimental Evaluation and Comparison of Explicit Force Control Strategies for Robotic Manipulators. In: *American Control Conference*, 1992, S. 758–765
- [VK93] VOLPE, R. ; KHOSLA, P.: A theoretical and experimental investigation of explicit force control strategies for manipulators. In: *IEEE Transactions on Automatic Control* 38 (1993), Nov, Nr. 11, S. 1634–1650
- [Whi85] WHITNEY, D.: Historical perspective and state of the art in robot force control. In: *Proceedings of IEEE International Conference on Robotics and Automation* Bd. 2, 1985, S. 262–268
- [WLZ07] WANG, Hesheng ; LIU, Yun-Hui ; ZHOU, Dongxiang: Dynamic Visual Tracking for Manipulators Using an Uncalibrated Fixed Camera. In: *IEEE Transactions on Robotics* 23 (2007), Jun, Nr. 3, S. 610–617
- [WLZ08] WANG, Hesheng ; LIU, Yun-Hui ; ZHOU, Dongxiang: Adaptive Visual Servoing Using Point and Line Features With an Uncalibrated Eye-in-Hand Camera. In: *IEEE Transactions on Robotics* 24 (2008), Aug, Nr. 4, S. 843–857
- [WMS04] WU, Zhongcheng ; MENG, Ming ; SHEN, Fei: Interaction force measurement of robotic manipulator based on 12DOF force sensor. In: *Proceedings of International Conference on Information Acquisition*, 2004, S. 240–243
- [WS07] WINKLER, A. ; SUCHY, J.: Dynamic force/torque measurement using a 12DOF sensor. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2007, S. 1870–1875

- [WSN85] WEISS, L.E. ; SANDERSON, A.C. ; NEUMAN, C.P.: Dynamic visual servo control of robots: An adaptive image-based approach. In: *Proceedings of IEEE International Conference on Robotics and Automation* Bd. 2, 1985, S. 662–668
- [WSN87] WEISS, L. ; SANDERSON, A. ; NEUMAN, C.: Dynamic sensor-based control of robots with visual feedback. In: *IEEE Journal of Robotics and Automation* 3 (1987), Okt, Nr. 5, S. 404–417
- [YA94] YOSHIMI, B.H. ; ALLEN, P.K.: Active, uncalibrated visual servoing. In: *Proceedings of IEEE International Conference on Robotics and Automation* Bd. 1, 1994, S. 156–161
- [YC78] YAKIMOVSKY, Y. ; CUNNINGHAM, R.: A system for extracting three-dimensional measurements from a stereo pair of TV cameras. In: *Computer Graphics and Image Processing* 7 (1978), Nr. 2, S. 195–210
- [Yos00] YOSHIKAWA, T.: Force control of robot manipulators. In: *Proceedings of IEEE International Conference on Robotics and Automation* Bd. 1, 2000, S. 220–226
- [ZH97] ZENG, Ganwen ; HEMAMI, Ahmad: An overview of robot force control. In: *Robotica* 15 (1997), Sep, Nr. 5, S. 473–482
- [ZZC08] ZHAO, Qingjie ; ZHANG, Liqun ; CHEN, Yunjiao: Online estimation technique for Jacobian matrix in robot visual servo systems. In: *3rd IEEE Conference on Industrial Electronics and Applications*, 2008, S. 1270–1275
- [ZZZJ08] ZOU, Yuanyuan ; ZHAO, Mingyang ; ZHANG, Lei ; JIANG, Chunying: Development of laser stripe sensor for automatic seam tracking in robotic tailored blank welding. In: *7th World Congress on Intelligent Control and Automation*, 2008, S. 3062–3066

Abbildungsverzeichnis

| | | |
|------|---|----|
| 1.1 | Übersicht über die Arbeit mit Kapitelangabe. | 14 |
| 1.2 | Visual Servoing Struktur (direct visual control) nach Weiss. | 17 |
| 1.3 | Dynamic look-and-move Struktur nach Weiss. | 18 |
| 1.4 | Klassifizierung bildbasierter Verfahren zur Kommandierung von Robotern. | 21 |
| 1.5 | Beispiel einer bildbasierten Positionieraufgabe. | 24 |
| 1.6 | Montage des Kraft-Momenten-Sensors (KMS) zwischen den Handflansch des Roboters und das Werkzeug. | 26 |
| 1.7 | Bolzen-in-Loch Aufgabe nach Mason. | 28 |
| 1.8 | Hybride Positions- und Kraftregelung nach Craig. | 29 |
| 1.9 | Kraft- und Bildinformationen in einem geschlossenen Re- gelkreis nach Nelson. | 30 |
| 1.10 | Der Konturfehler ist die Abweichung des tatsächlichen Pfa- des vom Referenzpfad (nach [CL07]) | 33 |
| 1.11 | Aufbau des Experimentalsystems an der Hochschule Darm- stadt. | 38 |
| 1.12 | Anregelzeit und Beschleunigung des Robotersystems bei sprungförmiger Änderung der Sollposition. | 39 |
| 2.1 | Die Position und Orientierung des TCP wird aus den Ge- lenkwinkeln $\theta_i, i = 1, \dots, 6$ berechnet. | 43 |
| 2.2 | Darstellung eines Antriebsstranges bestehend aus Getriebe, Motor und Antriebsriemen. | 44 |
| 2.3 | Modifizierter Rigid-Body Luenberger Observer. | 45 |
| 2.4 | Vergleich zwischen der Position aus der Robotersteuerung, Beobachtung und tatsächlicher Position des TCP bei hoher Massenträgheit | 48 |
| 2.5 | Vergleich zwischen der Position aus der Robotersteuerung, Beobachtung und tatsächlicher Position des TCP bei mitt- lerer Massenträgheit | 50 |

| | | |
|------|---|----|
| 2.6 | Vergleich zwischen der Position aus der Robotersteuerung, Beobachtung und tatsächlicher Position des TCP bei geringer Massenträgheit | 51 |
| 2.7 | Ausgang des Beobachters bei Weglassen des Beschleunigungssensors bzw. bei Weglassen des Positionssignales. . . . | 52 |
| 2.8 | Vergleich zwischen Geschwindigkeit und Beschleunigung durch Ableitung der Position. | 53 |
| 3.1 | Übersicht mit Kapitelangabe über die bildgestützte Konturverfolgung. | 56 |
| 3.2 | Systemaufbau zur Konturverfolgung mit vorausschauender Kameraanordnung. | 57 |
| 3.3 | Aufbau des Laser-Triangulationssensors. | 58 |
| 3.4 | Auswertung des Kamerabildes. | 59 |
| 3.5 | Einfluss der Zeitverzögerung τ_{IP} auf die Bildverarbeitung. . | 61 |
| 3.6 | Veränderlichkeit der Vorlauflänge im Kurvenbereich einer Kontur. | 63 |
| 3.7 | Rückläufige Abtastung einer Kontur im Kurvenbereich. . . | 64 |
| 3.8 | Einsortierung eines neuen Messwertes \mathbf{p}_{new} in den Merkmalspfad \mathbf{P} | 65 |
| 3.9 | Einsortierung der Messwerte an einer Ecke. | 67 |
| 3.10 | Einsortierung von Merkmalen an einer Ecke und bei falscher räumlicher Aufzeichnung entlang der Kontur. | 67 |
| 3.11 | Regelungsstruktur der bildbasierten prädiktiven Regelung zur Konturverfolgung. | 68 |
| 3.12 | Herstellen der geometrischen Relation zwischen Roboterpfad und Merkmalspfad. | 69 |
| 3.13 | Zeitverzögerung τ_{fil} , τ_{rob} durch Filterung und Übertragungsverhalten des Roboters. | 73 |
| 3.14 | Winkelfehler der Rotation um die y -Achse bei Konturverfolgung an einem Wellenprofil. | 74 |
| 3.15 | Ermittlung der translatorischen Korrektur. | 76 |
| 3.16 | Berechnung einer Tangente durch lineare bzw. quadratische Pfadapproximation in \mathbb{R}^3 | 78 |
| 3.17 | Anpassung des Werkzeuges an eine leicht gekrümmte Kontur bei $v_x = 60mm/sec$ | 81 |
| 3.18 | Filterung der Merkmalsposition durch quadratische Regression, lineare Regression und Mittelwertfilterung. | 82 |
| 3.19 | Konturverfolgung mit $v_x = 600mm/sec$ | 83 |
| 3.20 | Automatische Geschwindigkeitskorrektur am Wellenprofil. . | 86 |

| | | |
|------|--|-----|
| 3.21 | Filterung im Bereich einer Ecke. | 87 |
| 3.22 | Einpassung eines Dreiecks an die 3D-Kontur zur Ermittlung einer Ecke. | 88 |
| 3.23 | Roboterpfad entlang einer Kontur mit Ecke. | 89 |
| 4.1 | Anbringung des Kraft-Momenten Sensors (KMS) am Roboter. | 91 |
| 4.2 | Übersicht mit Kapitelangabe über die Kraftregelung. | 92 |
| 4.3 | Positionsbasierte explizite Kraftregelung. | 93 |
| 4.4 | Anbringung der Beschleunigungssensorik am Roboter. | 98 |
| 4.5 | Anbringung der drei 2-Achs Beschleunigungssensoren. | 99 |
| 4.6 | Vergleich der Beschleunigung durch Messmethode 1-3. | 100 |
| 4.7 | Vergleich der Beschleunigungsmessung zwischen Methode 1 und 2. | 102 |
| 4.8 | Vergleich der Kraftkompensation nach Methode 1 und 2. | 104 |
| 4.9 | Vergleich der Beschleunigungsmessung von Methode 2 und 3. | 105 |
| 5.1 | Übersicht mit Kapitelangabe über die Verbindung bildgestützter Regelung mit Kraftregelung | 108 |
| 5.2 | Experimentalsystem zur Konturverfolgung mit Objektkontakt an nachgiebigen Werkstücken. | 109 |
| 5.3 | Globale und lokale Objektverformung durch Kontakt. | 110 |
| 5.4 | Einfluss der Kontaktkraft auf die Konturverfolgung an einem einseitig eingespannten Metallblech. | 111 |
| 5.5 | Vergleich der Werkzeugorientierung entlang der nachgiebigen Kontur. Durch die veränderte Kontaktkraft (roter Kasten) neigt sich das Werkzeug signifikant, sodass es nicht mehr senkrecht zur Kontur steht. | 112 |
| 5.6 | Differenzposition und Differenzkraft zur Berechnung der Federsteifigkeit. | 115 |
| 5.7 | Dynamische Kompensation der Verformung. | 117 |
| 5.8 | Torsion des Werkstückes bei Kontakt. | 118 |
| 5.9 | Neigung der Kontur durch Kontakt. | 118 |
| 5.10 | Filterung der Kraftmessung am bewegten Roboter. | 119 |
| 5.11 | Berücksichtigung der Filterverzögerung zur Kompensation. | 120 |
| 5.12 | Schaltbild der dynamischen Entkopplung. | 121 |
| 5.13 | Kompensation der Objektverformung mit Berücksichtigung der Dynamik bei Änderung der Kontaktkraft. | 122 |
| 5.14 | Eingangs- zu Ausgangsverhalten bei Anregung des Systems. | 125 |
| 5.15 | Identifizierte Parameter für Masse, Dämpfung und Steifigkeit in Abhängigkeit der Position entlang der Kontur. | 126 |

| | | |
|------|---|-----|
| 5.16 | Identifikation mit Beobachter. | 126 |
| 5.17 | Systemaufbau mit Wellenprofil zur Konturverfolgung mit Kontakt. | 127 |
| 5.18 | Kraftregelung mit Geschwindigkeitsvorsteuerung. | 129 |
| 5.19 | Berechnung der Vorsteuerung. | 130 |
| 5.20 | Vorsteuerung zur Konturverfolgung mit Kontakt. | 131 |
| 5.21 | Einfluss der Kompensation der Werkstückverformung auf die Kraftvorsteuerung. | 132 |
| 5.22 | Einfluss der Kompensation der Werkstückverformung bei paralleler Kraft- und Bildregelung. | 133 |
| A.1 | Berechnung einer Tangente durch lineare / quadratische Pfadapproximation in \mathbb{R}^3 | 142 |

Tabellenverzeichnis

| | | |
|-----|---|-----|
| 1.1 | Messbereiche der Kraft-Momenten-Sensoren | 41 |
| 3.1 | Richtungsdefinition zur bildgestützten Konturverfolgung . . | 57 |
| 4.1 | Parameter zur dynamischen Kraftkompensation [WS07]. . . | 96 |
| 4.2 | Dynamische Anteile der Kräfte und Momente [WS07]. . . . | 97 |
| 5.1 | Aufgabendefinition Experiment 1 | 128 |